Dr #1338

TREE-1081

MINI-1081

THERRP - A THERMODYNAMIC PROPERTIES PROGRAM

R. S. DEEDS

May 1977

EG&G Idaho, Inc.

IDAHO NATIONAL ENGINEERING LABORATORY

ENERGY RESEARCH AND DEVELOPMENT

IDAHO OPERATIONS OFFICE UNDER CONTRACT EY-76-C-07-1570

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Printed in the United States of America
Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22161
Price: Printed Copy \$5.50; Microfiche \$3.00

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

TREE-1081

DISTRIBUTED UNDER CATEGORY:
UC-66d
GE--Utilization Technology
TID-4500, R65

THERRP - A THERMODYNAMIC
PROPERTIES PROGRAM

EG&G Idaho, Inc.
IDAHO NATIONAL ENGINEERING LABORATORY
Idaho Falls, Idaho 83401

NOTICE
This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

By

R. S. Deeds

Date Published - May 1977

Prepared for the

ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
DIVISION OF GEOTHERMAL ENERGY AND IDAHO OPERATIONS OFFICE
Under Contract No. EY-76-C-07-1570

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED
28

ACKNOWLEDGEMENT

The information in this document is based on Starling's modification to the Benedict-Webb-Rubin equation of state. Dr. Starling is a professor of chemical engineering at the University of Oklahoma and has worked under contract at the INEL. He played an active role in the development of the computer code presented in this report, and for his help and advice the author is very grateful.

ABSTRACT

This report documents the computer program THERPP, a program that calculates the thermodynamic properties of light hydrocarbons and mixtures of light hydrocarbons. A specific pressure-temperature or pressure-enthalpy grid is input to obtain properties in the desired region.

THERPP is a modification of the program HSGC⁽¹⁾. Thermodynamic properties are calculated using Starling's modification to the Benedict-Webb-Rubin equation of state.

	<u>Page</u>
ACKNOWLEDGEMENT	i
ABSTRACT	ii
1.0 INTRODUCTION	1
2.0 USER PACKAGE	5
2.1 Inputs	5
2.2 Outputs	12
3.0 DOCUMENTATION	22
3.1 SBWR	22
3.2 CVHCAP	26
3.3 DØMØUT	26
3.4 ECHØ	29
3.5 EDIT	30
3.6 PBPM	31
3.7 PDPBPM	32
3.8 PDPM	34
3.9 PRØPS	35
3.10 PUNØUT	36
3.11 TDPBPM	38
3.12 VAPRES	40
3.13 VATEMP	42
3.14 VLREGM	45
3.15 BLØCK DATA	46
4.0 REFERENCES.	48
APPENDIX A - Input Summary	A-i
APPENDIX B - Program Hierarchy	B-i
APPENDIX C - Program Listing and Sample Run	C-i

1.0 INTRODUCTION

The computer program THERPP consists of a modification to the program HSGC⁽¹⁾ and was designed to compute thermodynamic properties of light hydrocarbons and mixtures of light hydrocarbons. Properties can be generated in the form of either a pressure-temperature grid or a pressure-enthalpy grid. In its current form, THERPP will handle grid sizes up to 30x30, for mixtures with up to 15 components. Changing the program to accept larger grid sizes or mixtures with greater than 15 components would require only minor changes, centered around the dimension statements.

THERPP utilizes Starling's modification to the Benedict-Webb-Rubin equation of state⁽²⁾ to evaluate the thermodynamic properties. The original Benedict-Webb-Rubin equation of state can be expressed as shown below:

$$P = \frac{Rt}{v} + \frac{RTB-A-C/T^2}{v^2} + \frac{RTb^1-a^1}{v^3} + \frac{a^1\alpha^1}{v^6} + \frac{c^1}{v^3T^2} \left(1 + \frac{\gamma^1}{v^2}\right) \exp\left(-\frac{\gamma^1}{v^2}\right)$$

where P is pressure, R is the ideal gas constant, T is temperature, v is specific volume, and A, B, C, a¹, b¹, c¹, α¹ and γ¹ are required constants. Starling's modification results in the slightly more complex equation below:

$$P = \rho RT + (B_0 RT - A_0 - \frac{C_0}{T^2} + \frac{D_0}{T^3} - \frac{E_0}{T^4}) \rho^2 + (bRT - a - \frac{d}{T}) \rho^3 + \alpha(a + \frac{d}{T}) \rho^6 + \frac{c \rho^3}{T^2} (1 + \gamma \rho^2) \exp(-\gamma \rho^2)$$

where again P is pressure, R is the ideal gas constant, T is temperature, ρ is density, and A₀, B₀, C₀, D₀, E₀, a, b, c, d, α and γ are required constants.

This equation is a pressure explicit equation (ie. pressure is defined as a function of temperature and density) requiring eleven equation parameters, as compared with the eight parameters used in the Benedict-Webb-Rubin equation of state. The additional terms in Starling's modified equation make it possible to predict properties at reduced temperatures as low as $T_r = 0.3$ and reduced densities as great as $\rho_r = 3.0$. It therefore can be used for light hydrocarbons in the compressed liquid region in addition to higher temperature regions.

The fluids for which the program has been tested and for which the eleven equation parameters have been calculated are methane, ethane, propane, isobutane, n-butane, isopentane, n-pentane, n-hexane, n-heptane, n-octane, ethylene, propylene, nitrogen, carbon dioxide, and hydrogen sulfide. For problems involving one of these fluids as a single component system, the eleven equation parameters are treated as inputs to the program. For mixture calculations a set of equations have been derived to generate the eleven equation parameters for each of the fluids in the mixture. These equations use the critical constants of the fluid to calculate the equation parameters and were derived from data for the fluids mentioned above.

For calculations in the immediate vicinity of the system's critical point, the results should be monitored very closely. This is especially important for dew and bubble point calculations. As a rough rule of thumb, the dew and bubble point calculations are only reliable for pressures less than 95% of the critical pressure. If an input pressure is too near the critical point, such that the dew or bubble point search fails, a message is printed as a warning to the user, and execution continues.

The program was written in Fortran IV and has been used on both the IBM 360 and CDC 7600/CYBER 70 MODEL 76 computers. The IBM 360 version was written in double precision to obtain satisfactory convergence for the various iteration loops. The CDC version, due to the larger word size, was left in single precision. Other inherent differences between the two computer systems required that some additional syntax changes be made between the two versions, however these alterations do not affect the calculated results.

In order to be able to run the program on either of the two computers, an additional non-standard Fortran IV procedure was required. A special character may be used in Column 1 of a source card to identify it as unique to either the IBM or CDC computers. An "A" in Column 1 defines a card used by the IBM computer in the double precision mode. A "B" in Column 1 defines a card used by the CDC computer in the single precision mode. This notation has been included in all of the listings presented in this report. It is therefore necessary to modify the source deck by eliminating the character in Column 1 on some cards and discarding others, depending on which version is desired.

Additional work schedule at this time involves extending the list of possible fluids to include ammonia, water, and several of the fluorocarbons. This will require that suitable estimates for the various input parameters be defined, and may require slight modifications to the equation of state. Also scheduled is the development of a plotter routine to generate either a pressure-temperature, a pressure-enthalpy, or a temperature-entropy plot showing constant property lines for all of the thermodynamic properties calculated.

The remainder of the documentation presented herein consists first of a users package containing a detailed discription of the necessary input parameters and a discription of the output. This is followed by a discription of the modifications and additions made to HSGC. A complete program listing with sample output is included in the appendices.

2.0 USER PACKAGE

This section presents in detail the necessary inputs needed to run the program THERPP, along with a description of the output. A sample input deck is shown, along with the resulting output.

2.1 Inputs

The necessary input cards and the format used for each card is shown below along with a description of each individual input parameter. Where appropriate, recommended values for the input parameters are included. For easy reference, the following has also been summarized in Appendix A.

Card 1: IGRID, NC, NPHASE, ITNM, IPRNT, IDMPRT, IPUNCH, IPNPRT
(8I5)

IGRID defines the type of grid desired.

IGRID = 0 implies a pressure-enthalpy grid.

IGRID = 1 implies a pressure-temperature grid.

NC is the number of components in a given mixture.
($1 \leq NC \leq 15$)

NPHASE is the minimum number of iterations in PHASE.
NPHASE = 3 for $NC \leq 4$. NPHASE = 4 for $NC > 4$.

ITNM is the maximum number of iterations in PHASE. ITNM = 30.

IPRNT is a debug write control. If IPRNT = 0 or blank the program prints debug information concerning the progress of the program. Anything else suppresses this output.

IDMPRT is a print flag to produce detailed 2-phase information. If IDMPRT = 0, normal output is printed. If IDMPRT equals anything but zero or blank, a detailed description of the 2-phase region is printed. (See Section 2.2 for further details.)

IPNPRT is a print flag used to print on the line printer the information that will be sent to the card punch. If IPNPRT = 1, this output is printed. Anything else results in this output being suppressed. It should be noted that if IPNPRT = 1 and IPUNCH equals anything but one, the output is printed by the line printer, but the cards are not punched. It should also be noted that if

cards are desired (IPUNCH = 1), IPNPRT must be set to one.

Card 2: DMAX, EPSD, DPSV, FUGERR, EPSS, STEP, ENCØR (8F10.6)

DMAX is the maximum liquid molar density anticipated for the grid. This value is used as an upper bound for the iteration in subroutine DENTE. For most hydrocarbon mixtures, the recommended value is 2.0.

EPSD is the closure tolerance for the density search in DHFP. EPSD = 0.5E-4

DPSV is the closure tolerance for the fugacity calculations. FUGERR = 0.2E-3

EPSS is the closure tolerance for the subroutine SEARCH. EPSS = 0.2E-3

STEP is the false position initial step size in SEARCH. STEP = 1.0002

ENCØR is a correction factor for the enthalpies calculated in the program to make them consistent with published tables. Different authors use different datum states for enthalpy evaluation, and therefore the enthalpies in one set of tables may vary by a constant value from the enthalpies reported in a second set of tables. ENCØR is used to adjust the enthalpies calculated in this program by this constant when necessary. The datum state used by SBWR is for the mixture at 0°R and 0 psia.

Card 3: TPHIN, (ISWT(I), I=1,6) (8F10.6)

TPHIN is an initial temperature guess used for the pressure-enthalpy grid calculations in °R. TPHIN corresponds to the minimum temperature in the grid which occurs at the lowest grid pressure and the lowest grid enthalpy. Accuracy of this guess typically needs only to be ±50% of the actual grid temperature.

ISWT(I) is a print flag for the Ith iteration loop in subroutine ITER. Anything but zero or blank produces detailed output concerning the progress of the iteration. For a detailed description of ITER, see Reference 3.

Card 4: (CØMP(J,I), J = 1,4), IDCØM(I), CMW(I), TC(I), ACF(I),
CD(I), PC(I), TB(I)
(4A4,I4,6F10.6)
(Cards number 4 and 5 must be repeated for each component)

CØMP(J,I) is a 16 character alphanumeric variable used to store the name of component I.

IDCØM(I) is an identification number for component I.

CMW(I) is the molecular weight of component I.
(See Table 2.1-1)

TC(I) is the critical temperature of component I in °F.
(See Table 2.1-1)

CD(I) is the critical density of component I in mole/ft³.
(See Table 2.1-1)

PC(I) is the critical pressure of component I in psia.
(See Table 2.1-1)

TB(I) is the boiling point of component I at one atmosphere pressure in °F. (See Table 2.1-1)

Card 5: (CI(I,J), J = 1,7)
(2F9.6, 4E13.6, F10.6)
(Cards number 4 and 5 must be repeated for each component)

CI(I,J) are ideal gas polynomial coefficients used in the evaluation of enthalpy and entropy. The values of (CI(I,J) were obtained from Reference 4 and have been reproduced in Table 2.1-2 for the fluids of interest in this report.

Card 6: (A(I,J), J = 1, NPARM) (NPARM is set equal to 11 in BLOCK DATA
(6F13.6)
(Card number 6 is used only for NC = 1)

A(I,J) are the eleven equation coefficients in the modified BWR equation of state for component number I. A(I,1) through A(I,6) are entered on the first card and A(I,7) through A(I,11) are entered on the second card. Values for A(I,J) were taken from Reference 2 and have been reproduced in Table 2.1-3

Card 7: (CKIJ(I,J), J = IJ,NC) (IJ = I+1, I = 1,NC)
(8F10.6)
(Card number 7 is used only for NC>1)

CKIJ(I,J) is an interaction parameter between any two different components. Values for CKIJ(I,J) were taken from Reference 2 and have been reproduced in Table 2.1-4. CKIJ(I,J) is input on a series of cards, whose total number is equal to NC-1. The first card contains the interaction parameters between component 1 and components 2 through NC. The second card contains the interaction parameters between component 2 and components 3 through NC. This pattern continues until the final card is reached, which contains the interaction parameter between component NC-1 and component NC.

TABLE 2.1-1

Physical Properties of Pure Materials

Compound	Critical Pres. (psia)	Critical Temp. (°F)	Critical Density (lb-mole/ft ³)	Molecular Weight	Acentric Factor	Boiling Temp. @ Atmos. P. (°F)
Methane	673.1	-116.43	0.6274	16.042	0.013	-258.7
Ethane	709.8	90.03	0.4218	30.068	0.1018	-127.5
Propane	617.4	206.13	0.3121	44.094	0.157	-43.7
i-Butane	529.1	274.96	0.2373	58.120	0.183	10.9
n-Butane	550.7	305.67	0.2448	58.120	0.197	31.1
i-Pentane (2-Methylbutane)	483.0	369.00	0.2027	72.146	0.226	82.1
n-Pentane	489.5	385.42	0.2007	72.146	0.252	96.9
n-Hexane	440.0	453.45	0.1696	86.172	0.302	155.7
n-Heptane	396.8	512.85	0.1465	100.198	0.353	209.2
n-Octane	362.1	563.79	0.1284	114.224	0.412	258.2
Ethylene	742.1	49.82	0.5035	28.050	0.101	-154.7
Propylene	667.0	197.40	0.3449	42.080	0.150	-53.9
Nitrogen	492.0	-232.60	0.6929	28.016	0.035	-320.4
Carbon Dioxide	1071.0	87.80	0.6641	44.010	0.210	-109.3
Hydrogen Sulfide	1306.0	212.70	0.6571	34.076	0.105	-75.3

TABLE 2.1-2

Compound	Ideal Gas Polynomial Coefficients (CI(I,J))						
	CI(I,1)	CI(I,2)	CI(I,3) ($\times 10^3$)	CI(I,4) ($\times 10^6$)	CI(I,5) ($\times 10^{10}$)	CI(I,6) ($\times 10^{14}$)	CI(I,7)
Methane	-5.58114	0.564834	-0.282973	0.417399	-1.525576	1.958857	-0.623373
Ethane	-0.76005	0.273088	-0.042956	0.312815	-1.389890	2.007023	0.045543
Propane	-1.22301	0.179733	0.066458	0.250998	-1.247461	1.893509	0.178189
i-Butane	13.28660	0.036637	0.349631	0.005361	-0.298111	0.538662	0.609350
n-Butane	29.11502	0.002040	0.434879	-0.081810	0.072349	-0.014560	0.829122
i-Pentane (2-Methylbutane)	27.17183	-0.031504	0.469884	-0.098283	0.102985	-0.029485	0.871908
n-Pentane	27.17183	-0.002795	0.440073	-0.086288	0.081764	-0.019715	0.736161
n-Hexane	32.03560	-0.023096	0.461333	-0.097402	0.103368	-0.030643	0.767792
n-Heptane	30.70117	-0.023143	0.460981	-0.098074	0.104752	-0.031340	0.711737
n-Octane	29.50114	-0.022402	0.459712	-0.098062	0.104754	-0.031355	0.664632
Ethylene	51.78893	0.020724	0.385431	-0.082721	0.092318	-0.029284	1.359786
Propylene	26.17773	0.044867	0.324263	-0.030604	-0.083732	0.188977	0.901280
Nitrogen	12.32674	3.199617	0.392786	-0.290069	1.090069	-1.387867	-4.938247
Carbon Dioxide	4.77805	0.114433	0.101132	-0.026494	0.034706	-0.013140	0.343357
Hydrogen Sulfide	-0.61782	0.238575	-0.024457	0.041067	-0.130126	0.144852	-0.45932

TABLE 2.1-3

SBWR Equation of State Parameters for Pure Components

Compound	A(I,1)	A(I,2)	A(I,3) ($\times 10^{-8}$)	A(I,4)	A(I,5)	A(I,6)	A(I,7)	A(I,8) ($\times 10^{-8}$)	A(I,9) ($\times 10^{-10}$)	A(I,10) ($\times 10^{-4}$)	A(I,11) ($\times 10^{-10}$)
Methane	0.723251	7520.29	2.71092	1.48640	0.925404	2574.89	0.468828	4.37222	1.07737	4.74891	3.01122
Ethane	0.826059	13439.3	29.5195	2.99656	3.11206	22404.5	0.909681	68.1826	25.747	70.2189	1468.19
Propane	0.964762	18634.7	79.6178	4.56182	5.46248	40066.4	2.01402	274.461	45.3708	1505.20	2560.53
i-Butane	1.87890	37264.0	101.413	7.11486	8.58663	47990.7	4.23987	406.763	85.3176	2168.63	8408.60
n-Butane	1.56588	32544.7	137.436	7.54122	9.14066	71181.8	4.00985	700.044	33.3159	3642.38	230.902
i-Pentane (2-Methylbutane)	1.27752	35742.0	228.430	11.7384	19.8384	204344.	6.16154	1290.83	142.115	3492.20	2413.26
n-Pentane	2.44417	51108.2	223.931	11.8593	16.6070	162185.	7.06702	1352.86	101.769	3885.21	3908.60
n-Hexane	2.66233	45333.1	526.067	14.8720	29.4983	434517.	9.70230	3184.12	552.158	3274.60	62643.3
n-Heptane	3.60493	77826.9	615.662	24.7604	27.4415	359087.	21.8782	3748.76	777.123	835.115	636.251
n-Octane	4.86965	81690.6	996.546	21.9888	10.5907	131646.	34.5124	6420.53	790.575	18590.6	3464.19
Ethylene	0.747945	12133.9	16.3203	2.27971	2.62914	15988.1	0.589158	40.9725	5.17563	90.3550	1.61706
Propylene	0.114457	6051.36	97.4762	4.07919	7.64114	81880.4	1.36532	294.141	70.5921	541.935	3412.50
Nitrogen	0.677022	4185.05	1.37936	1.10011	0.833470	1404.59	0.302696	0.844317	1.95183	3.11894	121.648
Carbon Dioxide	0.394117	6592.03	29.5902	1.64916	0.97443	5632.85	0.395525	27.4668	40.9151	5.99297	1.02898
Hydrogen Sulfide	0.297508	10586.3	21.1496	1.20447	2.53315	2.05110	0.165961	43.6132	4.86518	1.99731	3.93226

TABLE 2.1-4

Values of Interaction Parameters (CKIJ(I,J)) (CKIJ(I,J)X100)

	Methane	Ethylene	Ethane	Propylene	Propane	i-Butane	n-Butane	i-Pentane (2-Methylbutane)	n-Pentane	Hexane	Heptane	Octane	Nitrogen	Carbon Dioxide	Hydrogen Sulfide	
11	0.0	1.0	1.0	2.1	2.3	2.75	3.1	3.6	4.1	5.0	6.0	7.0	2.5	5.0	5.0	Methane
		0.0	0.0	0.3	0.31	0.4	0.45	0.5	0.6	0.7	0.85	1.0	7.0	4.8	4.5	Ethylene
			0.0	0.3	0.31	0.4	0.45	0.5	0.6	0.7	0.85	1.0	7.0	4.8	4.5	Ethane
				0.0	0.0	0.3	0.35	0.4	0.45	0.5	0.65	0.8	10.0	4.5	4.0	Propylene
					0.0	0.3	0.35	0.4	0.45	0.5	0.65	0.8	10.0	4.5	4.0	Propane
						0.0	0.0	0.08	0.1	0.15	0.18	0.2	11.0	5.0	3.6	i-Butane
							0.0	0.08	0.1	0.15	0.18	0.2	12.0	5.0	3.4	n-Butane
								0.0	0.0	0.0	0.0	0.0	13.4	5.0	2.8	i-Pentane (2-Methylbutane)
									0.0	0.0	0.0	0.0	14.8	5.0	2.0	n-Pentane
										0.0	0.0	0.0	17.2	5.0	0.0	Hexane
											0.0	0.0	20.0	5.0	0.0	Heptane
												0.0	22.8	5.0	0.0	Octane
													0.0	0.0	0.0	Nitrogen
														0.0	3.5	Carbon Dioxide
															0.0	Hydrogen Sulfide

Card 8: (Z(I), I = 1, NC)
(8F10.6)

Z(I) is the mole fraction of component I.

Card 9: NUM1, NUM2
(8I5)

NUM1 is the number of temperature grid lines in a pressure-temperature grid or the number of enthalpy grid lines in a pressure-enthalpy grid. (NUM1 $\leq$ 30)

NUM2 is the number of pressure grid lines in the grid.
(NUM2 $\leq$ 30)

Card 10: (GRID1(I), I = 1, NUM1)
(8F10.6)

GRID1(I) is the value of grid temperature I ($^{\circ}$ F) for a pressure-temperature grid or the value of grid enthalpy I (Btu/lb) for a pressure enthalpy grid.

Card 11: (GRID2(I), I = 1, NUM2)
(8F10.6)

GRID2(I) is the value of grid pressure number I (psia).

The input section for the program has been set up to allow for stacking more than one data set per run. The additional sets are merely placed in back of the first set. The program executes each data set separately until an end of file is reached, at which point execution terminates.

2.2 Outputs

The first section of output consists of an input card echo. In this section the input data cards are reproduced by the line printer in the same format as they appear on the cards. At the top of the page the columns are numbered from 1 to 80, representing the 80 columns on the cards. At the far left of the page, each card is numbered. An example of this section is shown in Figure 2.2-1. The example shows two data sets for a mixture of isobutane and propane.

13

[illegible]

Figure 2.2-1 Input Card Echo

Cards 1 through 12 make up the first data set, while the second set consists of cards 13 through 24.

The second section of output is the edit section. This is the first output section for any given data set. In this section, the input parameters are printed out as a check to make sure that the inputs to the program were entered correctly. The variable names and their associated values are shown. An example of this section is in Figure 2.2-2 for a pressure-enthalpy grid of an isobutane-propane mixture.

Following the edit section, the numerical results are presented. At the top of this section, a heading is printed which defines the type of grid being generated and the units used in the tables. Under the heading are the saturation properties calculated by the program. For a pressure-temperature grid the saturation properties are calculated at the given temperature grid values for temperatures below the critical temperature. For a pressure-enthalpy grid the saturation properties are calculated at the given pressure grid values for pressures below the critical pressure. For both grids, the saturated liquid properties are denoted by $M = -1$, while the saturated vapor properties are denoted by $M = 0$. An example of this section is shown in Figure 2.2-3 for a mixture with an isobutane mole fraction of .75 and a propane mole fraction of .25.

The fourth section of output contains the thermodynamic properties of the fluid at the specified grid points. For the pressure-temperature grid, the output is broken down into sections of constant temperature, with pressure varying. For the pressure-enthalpy grid, the output is broken down into sections of constant pressure,

*****EDIT THIS CASE*****

IGRID,NC,NPHASE,ITNM,IPRNT,IDMPRT,IPUNCH,IPNPRT
 0 2 3 20 0 0 0 1

DMAX,EPST,EPSTV,FUGERR,EPSS,STEP,ENCOR
 .200000E+01 .100000E-06 .200000E-06 .300000E-06 .100000E-04 .100010E+01 0.

TPHIN,(ISWT(I),I=1,6)
 450.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000

(COMP(J,I),J=1,4),IDCOM(I),CMW(I),TC(I),ACF(I),CD(I),PC(I),TB(I)

ISOBUTANE 1 58.1200 734.6500 .1830 .2373 529.0999 470.5900
 CI .132866E+02 .366370E-01 .349631E-03 .536100E-08 -.298111E-10 .538662E-14 .609350E+00

PROPANE 2 44.0940 665.8200 .1570 .3121 617.4000 415.9900
 CI -.122301E+01 .179733E+00 .664580E-04 .250998E-06 -.124746E-09 .189351E-13 .178189E+00

((A(I,J),J=1,NPARM),I=1,NC)
 .195878E+01 .370810E+05 .119981E+11 .879762E+01 .105227E+02 .871019E+05 .466890E+01
 .563869E+11 .937704E+12 .162646E+08 .428647E+14 .147970E+01 .261006E+05 .634021E+10
 .515827E+01 .598999E+01 .441979E+05 .209025E+01 .231486E+11 .398203E+12 .713464E+07
 .204110E+14

CKIJ(I,J)
 0 .30000E-02
 .30000E-02 0.

COMPONENT MOLE FRACTIONS
 ISOBUTANE .75000
 PROPANE .25000

NUM1 AND NUM2 5 5

TEMPERATURE OR ENTHALPY GRID LINES
 -60.00 20.00 100.00 180.00 260.00

PRESSURE GRID LINES
 50.00 200.00 400.00 700.00 1000.00

Figure 2.2-2 Input Edit

PRESSURE - ENTHALPY GRID

TEMPERATURES IN F
 PRESSURES IN PSIA
 SPECIFIC VOLUMES IN CUBIC FEET PER POUND
 ENTHALPIES IN BTU/LB
 ENTROPIES IN BTU/LB-R
 SPECIFIC HEATS IN BTU/LB-R
 M=-1 IMPLIES LIQUID REGION
 M=0 IMPLIES VAPOR REGION
 M=2 IMPLIES 2-PHASE REGION

SATURATED PROPERTIES

TF	P	V	H	S	CP	CV	M
67.26	50.000	1.88064	123.01	1.2177	.4151	0.0000	0
55.30	50.000	.02885	-26.60	.9282	.5427	0.0000	-1
165.95	200.000	.46884	151.84	1.2252	.5432	0.0000	0
156.66	200.000	.03358	33.96	1.0337	.6756	0.0000	-1
228.36	400.000	.20188	163.74	1.2261	.7837	0.0000	0
222.26	400.000	.04228	86.12	1.1118	1.2722	0.0000	-1

Figure 2.2-3 Saturated Properties

with enthalpy varying. Grid points lying in the liquid region are denoted by $M = -1$, in the vapor region by $M = 0$, and in the 2-phase region by $M = 2$. In the standard form of the output, grid point properties lying in the 2-phase region represent a weighted average between the vapor properties and the liquid properties. Figure 2.2-4 shows an example of this section for the isobutane-propane mixture used previously.

It is possible to obtain a more detailed discription of the grid points falling in the 2-phase region by setting the input variable IDMPRT equal to 1. This produces output for 2-phase grid points in the format shown in Figure 2.2-5. This example shows the output for a pressure of 50 psia and an enthalpy of 20 Btu/lb for the isobutane-propane mixture. At the top of this section the quality of the vapor-liquid mixture is expressed as the weight percent of vapor present and the weight percent of liquid present. Under this the composition of both the vapor and liquid is given. Finally the properties of the liquid and the properties of the vapor are given, along with the net properties of the mixture. The net properties are the same as the values printed when IDMPRT is set to anything but 1 in the input deck.

The final section of output was written expressly to generate cards for the property tab decks in the computer program GEOSYS⁽³⁾. This section is accessed only for pressure-enthalpy grids and only when IPNPRT is set equal to 1. If IPUNCH is set equal to anything but 1, this output is sent only to the line printer and no cards are punched. If IPUNCH is set equal to 1, this output is sent to both the line printer and the card punch. An example of this section

GRID PROPERTIES

TF	P	V	H	S	CP	CV	M
-8.89	50.000	.02691	-60.00	.8590	.5004	0.0000	-1
60.29	50.000	.59714	20.00	1.0233	.5136	0.0000	2
66.04	50.000	1.57048	100.00	1.1763	.4462	0.0000	2
191.66	50.000	2.44423	180.00	1.3165	.4794	0.0000	0
345.53	50.000	3.09525	260.00	1.4254	.5606	0.0000	0

TF	P	V	H	S	CP	CV	M
-8.23	200.000	.02719	-60.00	.8512	.5060	0.0000	-1
135.10	200.000	.03244	20.00	1.0143	.6334	0.0000	-1
162.53	200.000	.26836	100.00	1.1454	.6152	0.0000	2
216.82	200.000	.54891	180.00	1.2598	.5396	0.0000	0
360.18	200.000	.73759	260.00	1.3769	.5839	0.0000	0

TF	P	V	H	S	CP	CV	M
-10.11	400.000	.02692	-60.00	.8571	.5007	0.0000	-1
135.17	400.000	.03200	20.00	1.0105	.6204	0.0000	-1
223.63	400.000	.06995	100.00	1.1371	1.0220	0.0000	2
250.40	400.000	.23078	179.99	1.2503	.6872	0.0000	0
376.98	400.000	.34440	260.00	1.3541	.6239	0.0000	0

TF	P	V	H	S	CP	CV	M
-12.06	700.000	.02673	-60.00	.8531	.4973	0.0000	0
134.98	700.000	.03155	20.00	1.0069	.6065	0.0000	0
244.64	700.000	.04265	100.00	1.1296	1.0300	0.0000	0
300.87	700.000	.10264	180.00	1.2388	1.1169	0.0000	0
400.85	700.000	.17842	260.00	1.3380	.7014	0.0000	0

TF	P	V	H	S	CP	CV	M
-13.81	1000.000	.02660	-60.00	.8498	.4952	0.0000	0
134.52	1000.000	.03118	20.00	1.0040	.5963	0.0000	0
250.98	1000.000	.04008	100.00	1.1263	.8393	0.0000	0
330.58	1000.000	.06985	180.00	1.2328	1.0162	0.0000	0
422.32	1000.000	.11726	260.00	1.3289	.7742	0.0000	0

Figure 2.2-4 Grid Properties

PRESSURE = 50.000 ENTHALPY = 20.000
 WT.% VAPOR = .30056 WT.% LIQUID = .69944

	LIQUID (WT.%)	VAPOR (WT.%)
ISOBUTANE	.84936	.67900
PROPANE	.15064	.32100

	TF	P	V	H	S	CP	CV	M
LIQUID PROPERTIES	60.29	50.000	.02936	-24.39	.9256	.5536	0.0000	2
VAPOR PROPERTIES	60.29	50.000	1.91843	123.29	1.2509	.4205	0.0000	2
NET PROPERTIES	60.29	50.000	.59714	20.00	1.0233	.5136	0.0000	2

Figure 2.2-5 Detailed 2-phase Output

is shown in Figure 2.2-6 for the previously used isobutane-propane mixture. The output for each property is read from right to left starting at both the lowest pressure and lowest enthalpy grid lines, incrementing pressure first. The format used to generate the table is:

```
FORMAT((5X,2H& ,5(E12.4,1H, )))
```

PUNCH OUTPUT

TEMPERATURE

E	-.8891E+01,	-.8230E+01,	-.1011E+02,	-.1206E+02,	-.1381E+02,
E	.6029E+02,	.1351E+03,	.1352E+03,	.1350E+03,	.1345E+03,
E	.6604E+02,	.1625E+03,	.2236E+03,	.2446E+03,	.2510E+03,
E	.1917E+03,	.2168E+03,	.2504E+03,	.3009E+03,	.3306E+03,
E	.3455E+03,	.3602E+03,	.3770E+03,	.4008E+03,	.4223E+03,

ENTROPY

E	.8590E+00,	.8612E+00,	.8571E+00,	.8531E+00,	.8498E+00,
E	.1023E+01,	.1014E+01,	.1011E+01,	.1007E+01,	.1004E+01,
E	.1176E+01,	.1145E+01,	.1137E+01,	.1130E+01,	.1126E+01,
E	.1316E+01,	.1270E+01,	.1250E+01,	.1239E+01,	.1233E+01,
E	.1426E+01,	.1377E+01,	.1354E+01,	.1338E+01,	.1329E+01,

SPECIFIC VOLUME

E	.2691E-01,	.2719E-01,	.2692E-01,	.2673E-01,	.2660E-01,
E	.5971E+00,	.3244E-01,	.3200E-01,	.3155E-01,	.3118E-01,
E	.1570E+01,	.2684E+00,	.6995E-01,	.4265E-01,	.4008E-01,
E	.2444E+01,	.5489E+00,	.2308E+00,	.1026E+00,	.6985E-01,
E	.3095E+01,	.7376E+00,	.3444E+00,	.1784E+00,	.1173E+00,

CP

E	.5004E+00,	.5060E+00,	.5007E+00,	.4973E+00,	.4952E+00,
E	.5136E+00,	.6334E+00,	.6204E+00,	.6065E+00,	.5963E+00,
E	.4462E+00,	.6152E+00,	.1022E+01,	.1030E+01,	.8393E+00,
E	.4794E+00,	.5396E+00,	.6872E+00,	.1117E+01,	.1016E+01,
E	.5606E+00,	.5839E+00,	.6239E+00,	.7014E+00,	.7742E+00,

Figure 2.2-6 PUNCH OUTPUT

3.0 DOCUMENTATION

As was mentioned previously, THERPP consists of a modification to the program HSGC⁽¹⁾. The modification consists essentially of re-writing the main program and adding a number of additional subroutines. In this section, the new main program, plus the additional subroutines are documented. The documentation for each subroutine is arranged in alphabetical order based on the subroutine name. Variables used by the subroutines are classified as either input, output, or intermediate. The definitions of these variables are presented wherever they appear as output or intermediate variables. Also included is the subroutine's hierarchy, a brief functional description, and a subroutine listing. Documentation for those subroutines from HSGC that were not altered for THERPP have not been included in this section. The user should refer to Reference 1 for further information concerning these subroutines.

One additional subroutine is used in THERPP that is not documented here or in Reference 1. This is subroutine ITER, a general Newton type iteration scheme. A detailed description of ITER can be found in Reference 3.

3.1 Main Program (SBWR)

1. Hierarchy:

Calls ECHO, EDIT, ITER, ITERM, PBPM, PDPM, PROPS, PUNOUT, TDPBBM, VAPRES, VATEMP, VLREGM

2. Functional Description:

SBWR serves as an executive program controlling the sequence of calculations used to generate the property grids. SBWR controls the inputs to the program,

as well as calling the necessary subroutines to calculate the properties and to produce the necessary outputs.

3. Variables:

a) Input

1. All inputs to the program (see Section 2.1)
2. CØMMØN/CA/V
3. CØMMØN/CB/NPARM
4. CØMMØN/CC/B(2,11),CP
5. CØMMØN/CT/AK(15),R,MD
6. CØMMØN/CZ/CV
7. CØMMØN/CX/VØLM,HMM,SMM,CPMM,CVMM
8. CØMMØN/H/H,S

b) Output:

1. CPGRID(I,J) - Constant pressure specific heat at grid point(I,J).
2. CPSATL(I) - Constant pressure specific heat of saturated liquid.
3. CPSATV(I) - Constant pressure specific heat of saturated vapor.
4. CVGRID(I,J) - Constant volume specific heat at grid point(I,J).
5. CVSATL(I) - Constant volume specific heat of saturated liquid.
6. CVSATV(I) - Constant volume specific heat of saturated vapor.
7. HGRID(I,J) - Enthalpy at grid point (I,J).
8. HGUESS - Intermediate enthalpy value.
9. HP - Enthalpy grid lines for pressure-enthalpy grid.
10. HSATL(I) - Enthalpy of saturated liquid.
11. HSATV(I) - Enthalpy of saturated vapor.

12. P - Calculation pressure.
13. PBP - Bubble point pressure.
14. PDP - Dew point pressure.
15. PH - Pressure grid lines for pressure-enthalpy grid.
16. PT - Pressure grid lines for pressure-temperature grid.
17. QUAL - Vapor-liquid quality.
18. SGRID(I,J) - Entropy at grid point (I,J).
19. SSATL(I) - Entropy of saturated liquid.
20. SSATV(I) - Entropy of saturated vapor.
21. T - Calculation temperature.
22. TBP - Bubble point temperature.
23. TDP - Dew point temperature.
24. TGRID(I,J) - Temperature at grid point (I,J).
25. TGUSS - Intermediate temperature value.
26. TP - Temperature grid lines for pressure-temperature grid.
27. VGRID(I,J) - Specific volume at grid point (I,J).
28. VSATL(I) - Specific volume of saturated liquid.
29. VSATV(I) - Specific volume of saturated vapor.
30. XL(I) - Mole fraction of component I.
31. XLMW - Mixture molecular weight.
32. ZC(I) - Critical compressibility factor.

c) Intermediate:

1. $AL(J) = A(1,J)$.
2. $CD2 = CD(I) * CD(I)$
3. I - Loop counter.
4. IJ - Loop counter.

5. IØNE - Loop counter.
6. J - Loop counter.
7. NCMI = NC - 1.
8. NDUM - Iteration limit.
9. NHF - Number of enthalpy grid lines.
10. NPF - Number of pressure grid lines.
11. NTF - Number of temperature grid lines.
12. PCMAX - Maximum critical pressure.
13. SUMZ - Sum of all mole fractions.
14. TCMAX - Maximum critical tamperature.
15. TC2 - TC(I)**2.
16. TC3 - TC2*TC(I).

4. Algorithm:

a) Procedure:

SBWR1) Call ECHØ

SBWR2) Read IGRID,NC,NPHASE,ITNM,IPRNT,IDMPRT,IPUNCH,
IPNPRT. If end of file, stop.

SBWR3) Read DMAX,EPST,EPSTV,FUGERR,EPSS,STEP,ENCØR.

SBWR4) Read CØMP(I,J),IDCØM,CMW,TC,ACF,CD,PC,TB, and CI for
each component.

SBWR5) Convert TC and TB to °F. Calculate ZC.

SBWR6) If NC = 1, read A. Set CKIJ = 0.

SBWR7) If NC > 1, calculate A and read CKIJ.

SBWR8) Read Z,NUM1,NUM2,GRID1,GRID2,

SBWR9) Define NTF,NHF,NPF,TP,HP,PT,PH.

SBWR10) Normalize the mole fractions.

SBWR11) Call EDIT.

SBWR12) Write headings.

- SBWR13) If IGRID $\neq$ 1, calculate saturation temperatures at the given pressures.
- SBWR14) If IGRID = 1, calculate saturation pressures at the given temperatures.
- SBWR15) Calculate properties at the saturation conditions.
- SBWR16) If IGRID $\neq$ 1, calculate the pressure-enthalpy grid point properties.
- SBWR17) If IGRID = 1, calculate the pressure-temperature grid point properties. Go to SBWR19.
- SBWR18) If IPNPRT = 1, call PUNOUT for punched card output for GEOSYS.
- SBWR19) Go to SBWR2.

3.2 Subroutine CVHCAP

1. Heirarchy: Called by PRØPS

2. Functional Discription:

CVHCAP is a subroutine used to generate the specific heat at constant volume for the mixture being used. At this point CVHCAP is a dummy subroutine and merely returns a value of 0.0 when called. When a suitable equation is derived, it can be inserted into this subroutine and used to calculate C_v .

3.3 Subroutine DØMØUT

1. Heirarchy: Called by VLREGM

2. Functional Discription:

DØMØUT is used to produce the output for grid points lying in the 2-phase region. Two forms of output are available. If IDMPRT is set equal to anything but 1 in the input deck, the 2-phase grid point properties are presented as a weighted average between the vapor properties and the liquid properties. If IDMPRT is set equal

to 1, a more detailed discription of these grid points is presented (see Section 2.2).

3. Variables:

a) Input:

1. CØMMØN/CA/CØMP(4,15),X(15),Y(15),Z(15),DG,DL,HG,HL,SG,SL,TF,V.
2. CØMMØN/CC/CMW,CPL,CPG.
3. CØMMØN/CT/P,T,MD,NC.
4. CØMMØN/CZ/XLMW.
5. CØMMØN/CX/VØLM,HMM,SMM,CPMM,CVMM.
6. CØMMØN/F2/IGRID,IDMPRT.
7. CØMMØN/H/ENCØR

b) Output

1. Input variables COMP(4,15),CPMM,CVMM,HMM,MD,P,SMM,TF,VOLM.
2. CPGM - Constant pressure specific heat of vapor. (Btu/lb-R).
3. CPLM - Constant pressure specific heat of liquid. (Btu/lb-R).
4. CVGM - Constant volume specific heat of vapor. (Btu/hr-R).
5. CVLM - Constant volume specific heat of liquid. (Btu/hr-R).
6. HGM - Vapor enthalpy (Btu/lb).
7. HLM - Liquid enthalpy (Btu/lb).
8. LM - Weight percent of liquid present in 2-phase mixture.
9. SGM - Vapor entropy (Btu/lb-R).
10. SLM - Liquid entropy (Btu/lb-R).
11. VGM - Vapor specific volume (ft^3/lb).
12. VLM - Liquid specific volume (ft^3/lb).

13. VM - Weight percent of vapor present in 2-phase mixture.
14. XM - Weight percent of given component in the liquid.
15. YM - Weight percent of given component in the vapor.

c) Intermediate:

1. I - Loop counter and component identification number.
2. J - Loop counter.
3. XMW - Liquid molecular weight.
4. YMW - Vapor molecular weight.
5. ZMW - Mixture molecular weight.

4. Algorithm:

a) Procedure

- DØMØ1) Initialize ZMW,XMW,YMW,TF
- DØMØ2) If IDMPRT = 0 go to DØMØ10.
- DØMØ3) Write heading.
- DØMØ4) If IGRID = 0, write P-T heading. Go to DØMØ6.
- DØMØ5) If IGRID = 1, write P-H heading. Go to DØMØ6.
- DØMØ6) Calculate ZMW,XMW,YMW.
- DØMØ7) Calculate and write VM and LM.
- DØMØ8) Calculate and write XM and YM.
- DØMØ9) Calculate and write VLM,HLM,HGM,SLM,SGM,CPLM,CPGM,CVLM,CVGM,VLM,VGM. Write TF,P,MD. Return.
- DØMØ10) Write TF,P,VØLM,HMM,SMM,CPMM,CVMM,MD. Return.

3.4 Subroutine ECHØ

1. Heirarchy: Called by SBWR

2. Functional Discription:

Subroutine ECHØ is used to reproduce the input deck on the line printer in the same format used on the cards. This provides a quick means of checking the input deck if a problem arises.

3. Variables:

a) Input:

1. DATA - Alphanumeric variable containing the information on the input card.

b) Output:

1. DATA - Same as above.

c) Intermediate:

1. I - Loop counter.
2. ILINE - Counter to keep track of the number of cards read.
3. NIT - Input device number.
4. NØT - Output device number.

4. Algorithm:

a) Procedure:

- ECHØ1) Write heading.
- ECHØ2) Define NIT and NØT and initilize ILINE.
- ECHØ3) Read DATA. If end of file reached, go to ECHØ6.
- ECHØ4) Increment ILINE.
- ECHØ5) Write ILINE and DATA. Go to ECHØ3.
- ECHØ6) Backspace input file to beginning of the file. Return.

3.5 Subroutine EDIT

1. Hierarchy: Called by SBWR

2. Functional Description:

Subroutine EDIT provides a second check on the input deck.

The input parameter names are printed out along with their associated values as stored in the computer memory. If a mistake is made in setting up the input deck, it should show up in this output.

3. Variables:

a) Input:

1. All parameters input through the input deck.

b) Output:

1. All input parameters.

c) Intermediate:

1. I - Loop counter.
2. J - Loop counter.

4. Algorithm:

a) Procedure:

EDIT1) Write heading.

EDIT2) Write IGRID,NC,NPHASE,ITNM,IPRNT,IDMPRT,IPUNCH,IPNPRT.

EDIT3) Write DMAX,EPDS,EPST,FUGERR,EPSS,STEP,ENCOR.

EDIT4) Write TPHIN,ISWT.

EDIT5) Write COMP,IDCOM,CMW,TC,ACF,CD,PC,TB, and CI for each component.

EDIT6) Write A(I,J).

EDIT7) Write COMP and Z for each component.

EDIT8) Write NUM1 and NUM2.

EDIT9) Write GRID1.

EDIT10) Write GRID2.

EDIT11) Return.

3.6 Subroutine PBPM

1. Heirarchy: Called by SBWR.

Calls DEWBUB,PDPBPM.

2. Functional Discription:

PBPM is used to calculate the bubble point pressure when temperature is given for mixtures of 2 or more components. Subroutine PDPBPM is called to generate an initial guess of the bubble point pressure. Control is then sent to subroutine DEWBUB to determine the actual bubble point.

3. Variables:

a) Input:

1. COMMON/CJ/ZC(15),PC(15),TC(15).

2. COMMON/CT/P,T,NC.

3. COMMON/XYZ/PDPBPI.

b) Output:

1. AK(I) - Equilibrium constant for component I.

2. NTYPE - Flag to specify bubble point or dew point search in DEWBUB.

3. PF - Pressure sent to DEWBUB.

c) Intermediate:

1. I - Loop counter.

2. IFLAG - Dummy variable.

3. TR - Reduced temperature.

4. Algorithm:

a) Basis:

The equation used to calculate the equilibrium constants assumes that the mixture behaves as an ideal solution with liquid in equilibrium with an ideal gas. Under these restrictions, the equilibrium constant for component I can be expressed as the vapor pressure of component I divided by the total pressure. The vapor pressure of component I can be calculated from the generalized reduced vapor equation:

$$P_{\text{VAPOR}} = (P_C(I) * \text{EXP}\{2.3026 * [(16.25 - 73.85 * ZC(I) + 90 * ZC(I)**2) * ((TR - 1)/TR) - 10 * (-8.68 * (TR - 1.8 + 6.2 * ZC(I)**2))]\})$$

The equilibrium constant can then be calculated by dividing P_{VAPOR} by the total system pressure (P), or:

$$AK(I) = P_{\text{VAPOR}}/P$$

b) Procedure:

- PBPM1) Initialize P and NTYPE.
- PBPM2) Calculate AK(I) for each component.
- PBPM3) Call PDPBPM for initial bubble point pressure.
- PBPM4) Call DEWBUB to calculate actual bubble point pressure.
- PBPM5) Return.

3.7 Subroutine PDPBPM

1. Hierarchy: Called by PBPM, PDPM

2. Functional Description:

PDPBPM is used to generate an initial guess of the dew point and bubble point pressure at a given temperature for use in subroutines PBPM and PDPM.

3. Variables:

a) Input:

1. CØMMØN/CA/Z(15).
2. CØMMØN/CT/AK(15),P,NC.

b) Output:

1. PDPBPI - Initial estimate of pressure.

c) Intermediate:

1. I - Loop counter.
2. PBPI - Intermediate bubble point pressure.
3. PDPI - Intermediate dew point pressure.

4. Algorithm:

a) Basis:

Using the ideal equilibrium constants calculated in PBPM and PDPM, estimates of the bubble point and dew point pressures can be made. For a bubble point:

$$PBPI = AK(I) * Z(I) * P$$

and for a dew point:

$$PDPI = \left[\frac{Z(I)}{(AK(I) * P)} \right]^{-1}$$

These values are then averaged and the resulting pressure is returned to the calling subroutine.

b) Procedure:

- PDPB1) Initialize PBPI and PDPI
- PDPB2) Calculate PBPI and PDPI.
- PDPB3) Calculate PDPBPI.
- PDBB4) Return.

3.8 Subroutine PDPM

1. Hierarchy: Called by SBWR.

Calls DEWBUB, PDPBPM

2. Functional Description:

PDPM is used to calculate the dew point pressure when temperature is given for mixtures of two or more components. Subroutine PDPBPM is called to generate an initial guess of the dew point pressure. Control is then sent to subroutine DEWBUB to determine the actual dew point.

3. Variables:

- a) Input:

1. COMMON/CK/ZC(15), PC(15), TC(15).
2. COMMON/CT/P, T, NC.
3. COMMON/XYZ/PDPBPI

- b) Output:

1. AK(I) - Equilibrium constant for component I.
2. NTYPE - Flag to specify bubble point or dew point search in DEWBUB.
3. PF - Pressure sent to DEWBUB.

- c) Intermediate:

1. I - Loop counter.
2. IFLAG - Dummy variable.
3. TR - Reduced temperature.

4. Algorithm:

- a) Basis:

The equation used to calculate the equilibrium constants in PDPM is the same as the equation used in PBPM.

b) Procedure:

- PDPM1) Initialize P and NTYPE.
- PDPM2) Calculate AK(I) for each component.
- PDPM3) Call PDPBPM for initial dew point pressure.
- PDPM4) Call DEWBUB to calculate actual bubble point pressure.
- PDPM5) Return.

3.9 Subroutine PRØPS

1. Heirarchy: Called by SBWR

Calls BWRC,CVHCAP,DHFP,HSCP.

2. Functional Discription:

Subroutine PRØPS is called to evaluate grid point properties at points outside of the 2-phase dome. Coefficients for the Starling modified Benedict-Webb-Rubin equation of state are called from BWRC and sent to DHFP,HSCP, and CVHCAP where the properties are evaluated. The properties are then converted to units based on mass rather than moles. A print flag (FLAG) is also provided to print out the calculated properties when desired.

3. Variables:

a) Input:

- 1. CØMMØN/CA/TF
- 2. CØMMØN/CC/HC,SC,CP
- 3. CØMMØN/CT/DC,P,MD,NC,T
- 4. CØMMØN/CZ/XLMW,CV
- 5. CØMMØN/H/ENCØR
- 6. FLAG - Print flag. FLAG = 0 produces output.
Anything else supresses output.

b) Output:

1. Input variables TF,P, and MD.
2. CP - Constant pressure specific heat. (Btu/lb-°R)
3. CV - Constant volume specific heat. (Btu/lb-°R)
4. H - Enthalpy. (Btu/lb-°R)
5. S - Entropy. (Btu/lb-°R)
6. V - Specific volume. (ft³/lb)

4. Algorithm:

a) Procedure:

- PRØP1) If NC = 1, go to PRØP3.
- PRØP2) Call BWRC for equation of state coefficients.
- PRØP3) Call DHFP for density at given T and P.
- PRØP4) Call HSCP for remaining properties.
- PRØP5) Convert properties to mass basis.
- PRØP6) If FLAG = 0, write TF,P,V,H,S,CP,CV,MD.
- PRØP7) Return.

3.10 Subroutine PUNØUT

1. Heirarchy: Called by SBWR

2. Functional Discription:

PUNØUT is used to produce the punched card output for use in the computer program GEØSYS⁽³⁾. The output sent to the card punch is also sent to the line printer for easy reference. For additional information concerning the output, see Section 2.2.

3. Variables:

a) Input

1. COMMON/F1/TGRID(30,30),SGRID(30,30),VGRID(30,30)
CPGRID(30,30),CVGRID(30,30),IPUNCH,NPF,NHF.

b) Output:

1. All input variables.

c) Intermediate:

1. I - Loop counter.
2. J - Loop counter.

4. Algorithm:

a) Procedure:

- PUNØ1) Write heading.
- PUNØ2) Write TGRID.
- PUNØ3) Write SGRID.
- PUNØ4) Write VGRID.
- PUNØ5) Write CPGRID.
- PUNØ6) Return.

Note: All of the above write statements imply writing on the line printer. If IPUNCH = 1, the variables are also punched on cards.

3.11 Subroutine TDPBPM

1. Hierarchy: Called by: SBWR

Calls: ITER

2. Functional Discription:

TDPBPM is used to generate an estimate of the dew point temperature and the bubble point temperature for a given pressure when a pressure-enthalpy grid is being used for a mixture of two or more fluids. These estimates are returned to SBWR where they are used as initial guesses for the dew point and bubble point searches.

3. Variables:

a) Input:

1. COMMON/CA/Z(15).
2. COMMON/CK/ZC(15),PC(15),TC(15).
3. COMMON/CT/AK(15),T,NC,P.
4. COMMON/CZ/TB(15).
5. PHIII - Pressure at which dew point and bubble point temperatures are sought.

b) Output:

1. TDPI - Dew point temperature estimate.
2. TBPI - Bubble point temperature estimate.

c) Intermediate:

1. FLAG - Closure flag from ITER.
2. I - Component sequence number.
3. IZMAX - Sequence number of component with largest mole fraction.
4. P - Intermediate pressure.
5. PB - 1 atmosphere pressure, 14.696 psia.

6. PBPI - Intermediate bubble point pressure.
7. PDPI - Intermediate dew point pressure.
8. TR - Reduced temperature (T/TC).
9. ZA - Intermediate value.
10. ZB - Intermediate value.
11. ZMAX - Largest component mole fraction in the mixture.

4. Algorithm:

a) Basis:

Initially the component with the largest mole fraction is identified and its sequence identification number is noted along with its mole fraction. A first guess of both the bubble point (B.P.) and dew point (D.P.) temperature can then be calculated for this component from the equations:

$$ZB = \frac{\ln \left(\frac{PC}{PB} \right)}{\left(\frac{1}{TB} - \frac{1}{TC} \right)}$$

$$ZA = -\frac{ZB}{TB} + \ln \left(\frac{PB}{P} \right)$$

$$T = ZB/ZA$$

Using this temperature a B.P. Pressure and a D.P. pressure can be calculated from the equations:

$$PBPI = (AK(I) * Z(I) * P)$$

$$PDPI = \frac{1}{\left(\frac{Z(I)}{AK(I) * P} \right)}$$

These pressures are compared in ITER with the desired pressure (PHIII), and a new temperature guess is generated. The iteration continues until the pressures match, yielding the B.P. and D.P. temperatures. Since these temperatures are only estimates of the

actual B.P. and D.P. temperatures, the closure tolerance in ITER is set rather large (ie. 5.E-2).

b) Procedure:

- TDPB1) Determine which component has the largest mole fraction.
- TDPB2) Calculate initial temperature guess (T).
- TDPB3) Calculate B.P. pressure at T.
- TDPB4) Call ITER to compare PBPI with PHIII and to calculate new T. If within tolerances, go to TDPB5. If not go to TDPB3.
- TDPB5) $T_{BPI} = T$.
- TDPB6) Calculate D.P. pressure at T.
- TDPB7) Call ITER to compare PDPI with PHIII and to calculate new T. If within tolerances, go to TDPB8. If not go to TDPB6.
- TDPB8) $T_{DPI} = T$.
- TDPB9) Return.

3.12 Subroutine VAPRES

1. Hierarchy: Called by SBWR
Calls DHFP, HSCP

2. Functional Description:

Subroutine VAPRES is used to determine the saturation pressure at a given temperature for single component systems. An iterative search is used to locate this pressure. Closure is defined as the pressure at which the fugacity of the liquid is equal to the fugacity of the vapor. The fugacities are also used to update the pressure estimate. If the iteration does not close within 20 attempts, a message is printed to the user.

3. Variables:

a) Input:

1. COMMON/CC/CONV,NC,SC.
2. COMMON/CK/PC(15),TC(15).
3. COMMON/CT/R,T.
4. COMMON/CZ/TB(15).
5. COMMON/CY/HIDEAL,SIDEAL.

b) Output:

1. P - Saturation pressure.

c) Intermediate:

1. FUG - Vapor fugacity.
2. FUL - Liquid fugacity.
3. J - Loop counter.
4. MD - Phase flag. MD = 0 implies vapor. MD = -1 implies liquid.
5. PB - 14.696 psia.
6. ZA - Intermediate value in initial pressure guess.
7. ZB - Intermediate value in initial pressure guess.

4. Algorithm:

a) Basis:

The initial guess for the saturation pressure is calculated from the equations:

$$ZB = \frac{\ln(PC) - \ln(PB)}{\left(\frac{1}{TB} - \frac{1}{TC}\right)}$$

$$ZA = \ln(PB) + \frac{ZB}{TB}$$

$$P = \exp\left(ZA - \frac{ZB}{T}\right)$$

With this pressure and the given temperature, subroutines DHFP and HSCP are called to obtain values for HC, SC, HIDEAL, and SIDEAL assuming a liquid state exists and assuming a vapor state exists. From this the fugacity of the liquid (FUL) and the fugacity of the vapor (FUG) are calculated. If the fugacities are equal, the saturation pressure has been found and is returned to SBWR. If they are not equal the pressure estimate is updated using the equation:

$$P = P * \frac{FUL}{FUG}$$

and the iteration continues.

b) Procedure:

- VAPR1) Calculate initial pressure guess.
- VAPR2) Call DHFP and HSCP for MD = 0.
- VAPR3) Calculate FUG.
- VAPR4) Call DHFP and HSCP for MD = -1.
- VAPR5) Calculate FUL.
- VAPR6) If $(FUL - FUG)/(FUG) < .0001$, return.
- VAPR7) Update pressure estimate.
- VAPR8) Check for maximum number of iterations. If J less than 20, go to VAPR2.
- VAPR9) Write nonclosure message.
- VAPR10) Return.

3.13 Subroutine VATEMP

1. Hierarchy: Called by SBWR
Calls DHFP, HSCP
2. Functional Description:

Subroutine VATEMP is used to determine the saturation temperature at a given pressure for single component systems. An iterative search is used to locate this temperature. Closure is defined as the temperature at which the fugacity of the liquid is equal to the fugacity of the vapor. Updates to the temperature estimate are determined using the standard secant method. If the iteration does not close within 20 attempts, a message is printed to the user.

3. Variables:

a) Input:

1. COMMON/CC/CONV,HC,SC.
2. COMMON/CK/PC(15),TC(15).
3. COMMON/CT/R,P
4. COMMON/CZ/TB(15).
5. COMMON/CY/HIDEAL,SIDEAL

b) Output:

1. T - Saturation temperature.

c) Intermediate:

1. FUG - Vapor fugacity.
2. FUGAC(I) - Vapor fugacity for iteration number I.
3. FUL - Liquid fugacity.
4. FULAC(I) - Liquid fugacity for iteration number I.
5. I - Loop counter.
6. MD - Phase flag. MD = 0 implies vapor.
MD = -1 implies liquid.
7. PB - 14,696 psia.
8. SLOPE - Factor used for updating temperature estimate.

9. TEMP(I) - New guess for temperature.

10. ZA - Intermediate value in initial temperature guess.

11. ZB - Intermediate value in initial temperature guess.

4. Algorithm:

a) Basis:

The initial guess for the saturation temperature is calculated from the equations:

$$ZB = \frac{\ln(PC) - \ln(PB)}{\left(\frac{1}{TB} - \frac{1}{TC}\right)}$$

$$ZA = \frac{ZB}{TB} + \ln\left(\frac{PB}{P}\right)$$

$$T = ZB/ZA$$

With this temperature and the given pressure, subroutines DHFP and HSCP are called to obtain values for HC, SC, HIDEAL, and SIDEAL assuming a liquid state exists and assuming a vapor state exists. From this the fugacity of the liquid and the fugacity of the vapor are calculated. If the fugacities are equal, the saturation temperature has been found and is returned to SBWR. If they are not equal the temperature estimate is updated and the iteration continues. For the first update the temperature is increased by 1.0°F. All following updates are calculated from the following equations:

$$SLOPE = \frac{FUGAC(I) - FULAC(I) - FUGAC(I-1) + FULAC(I-1)}{TEMP(I) - TEMP(I-1)}$$

$$TEMP(I+1) = \frac{SLOPE * TEMP(I) - FUGAC(I) + FULAC(I)}{SLOPE}$$

b) Procedure:

VATE1) Calculate initial temperature guess.

VATE2) Call DHFP and HSCP for MD = 0.
 VATE3) Calculate FUG.
 VATE4) Call DHFP and HSCP for MD = -1.
 VATE5) Calculate FUL.
 VATE6) If $(FUL - FUG)/(FUG) < .0001$, return.
 VATE7) If I = 1, TEMP(2) = TEMP(1)+1.0, go to VATE2.
 VATE8) Calculate new temperature guess.
 VATE9) If I < 20, go to VATE2.
 VATE10) Write nonclosure message.
 VATE11) Return.

3.14 Subroutine VLREGM

1. Heirarchy: Called by SBWR

Calls DØMØUT,KI,PHASE

2. Functional Discription:

Subroutine VLREGM is called to calculate grid point properties lying in the two phase region for mixtures with two or more components. These properties are returned to SBWR and if flagged are sent to DØMØUT to be printed.

3. Variables:

a) Input:

1. CØMMØN/CA/DG,DL,HM,SM,V
2. CØMMØN/CC/CPM
3. CØMMØN/CZ/XLMW
4. CØMMØN/H/ENCØR
5. FLAG - Print flag. Anything but 0 supresses output.

b) Output:

1. CPM - Constant pressure specific heat of 2-phase mixture (Btu/lb-°R).
2. CVM - Constant volume specific heat of 2-phase mixture (Btu/lb-°R).
3. HMM - Enthalpy of 2-phase mixture (Btu/lb).
4. MD - Phase flag. MD = 2 implies 2-phase region.
5. SMM - Entropy of 2-phase mixture (Btu/lb-°R)
6. VØLM - Specific volume of 2-phase mixture (ft³/lb).

4. Algorithm:

a) Procedure:

- VLRG1) Call KI to calculate equilibrium constants.
VLRG2) Call PHASE to calculate 2-phase properties.
VLRG3) Convert mixture properties to mass basis.
VLRG4) If FLAG = 0, call DØMØUT to print results.
VLRG5) Return.

3.15 BLOCK DATA

The variables initialize in BLOCK DATA are shown below:

B(1,1) = 0.44369E0	B(1,7) = 0.070523E0
B(1,2) = 1.2844E0	B(1,8) = 0.50409E0
B(1,3) = 0.35631E0	B(1,9) = 0.030745E0
B(1,4) = 0.54498E0	B(1,10) = 0.073283E0
B(1,5) = 0.52863E0	B(1,11) = 0.00645E0
B(1,6) = 0.48401E0	
B(2,1) = 0.11545E0	B(2,7) = -0.044448E0
B(2,2) = 0.92073E0	B(2,8) = 1.3224E0
B(2,3) = 1.7087E0	B(2,9) = 0.17943E0
B(2,4) = -0.2709E0	B(2,10) = 0.46349E0
B(2,5) = 0.34926E0	B(2,11) = 0.022143E0
B(2,6) = 0.75413E0	
ISWT(1) = 0.0	ISWT(4) = 0.0
ISWT(2) = 0.0	ISWT(5) = 0.0
ISWT(3) = 0.0	ISWT(6) = 0.0

EPS(1) = 5.E-5
EPS(2) = 5.E-5
EPS(3) = 5.E-5

MINX(1) = 0.0
MINX(2) = 0.0
MINX(3) = 0.0

MAXX(1) = 2.E3
MAXX(2) = 2.E3
MAXX(3) = 2.E3

PCNT1(1) = 5.E-2
PCNT1(2) = 5.E-3
PCNT1(3) = 5.E-3

PCNTN(1) = 1.E0
PCNTN(2) = 1.E0
PCNTN(3) = 1.E0

ICNT(1) = 0.0
ICNT(2) = 0.0
ICNT(3) = 0.0

ITF(1) = 1.0
ITF(2) = 1.0
ITF(3) = 1.0

XNAM(1) = TGUESS
XNAM(2) = TDPM
XNAM(3) = TBPM

YNAM(1) = HGUESS
YNAM(2) = PDPM
YNAM(3) = PBPM

MI = 9.E0

NE = 19.E0

EPS(4) = 5.E-5
EPS(5) = 5.E-2
EPS(6) = 5.E-2

MINX(4) = 0.0
MINX(5) = 0.0
MINX(6) = 0.0

MAXX(4) = 2.E3
MAXX(5) = 2.E3
MAXX(6) = 2.E3

PCNT1(4) = 5.E-3
PCNT1(5) = 1.E-3
PCNT1(6) = 1.E-3

PCNTN(4) = 1.E0
PCNTN(5) = 1.E0
PCNTN(6) = 1.E0

ICNT(4) = 0.0
ICNT(5) = 0.0
ICNT(6) = 0.0

ITF(4) = 1.0
ITF(5) = 1.0
ITF(6) = 1.0

XNAM(4) = TVLR
XNAM(5) = TBPGE
XMAN(6) = TDPGE

YNAM(4) = HVLR
YNAM(5) = PRES
YNAM(6) = PRES

4.0 REFERENCES

1. K. E. Starling, L. W. Fish, HSGC, A Mixture Thermodynamic Properties Computer Program, reports number ORO-4944-2 and ORO-4918-5, University of Oklahoma, Norman, Oklahoma, December, 1975.
2. K. E. Starling, Fluid Thermodynamic Properties for Light Petroleum Systems, Gulf Publishing Company, Houston, Texas, 1973.
3. I. J. Ingvarsson, W. W. Madsen, Determination of the 5 MW Gross Nominal Design Case Binary Cycle for Power Generation at Raft River, Idaho, TREE-1039, December, 1976.
4. C. A. Passut and R. P. Danner, "Correlation of Ideal Gass Enthalpy, Heat Capacity, and Entropy", I&EC Process Design and Development, Vol. 11, October, 1972.

APPENDIX A

INPUT SUMMARY

INPUT PARAMETERS

Card #	Format	Fortran Variable
1	8I5	IGRID,NC,NPHASE,ITNM,IPRNT, IDMPRT,IPUNCH,IPNPRT
2	8F10.6	DMAX,EPDS,EPVS,FUGERR,EPSS STEP,ENCØR
3	8F10.6	TPHIN,(ISWT(I),I=1,6)
4*	4A4,I4,6F10.6	(CØMP(J,I),J=1,4),IDCØM(I),CMW(I), TC(I),ACF(I),CD(I),PC(I),TB(I)
5*	2F9.6,4E13.6,F10.6	(CI(I,J),J=1,7)
6**	6F13.6	(A(I,J),J=1,NPARM)
7***	8F10.6	(CKIJ(I,J),J=IJ,NC)
8	8F10.6	(Z(I),I=1,NC)
9	8I5	NUM1,NUM2
10	8F10.6	(GRID1(I),I=1,NUM1)
11	8F10.6	(GRID2(I),I=1,NUM2)

- * Cards number 4 and 5 must be repeated for each component
- ** Card number 6 used only for single component systems
- *** Card number 7 used only for NC>1

Variable Name	Definition
IGRID	Grid flag: IGRID=0→PH GRID IGRID=1→PT GRID
NC	Number of components in mixture
NPHASE	Minimum number of iterations in PHASE
ITNM	Maximum number of iterations in SEARCH, DHFP, and FLASH (ITNM≤30)
IPRNT	Debug write control; anything but blank or zero produces printout
IDMPRT	Print flag for additional 2-phase infor- mation, IDMPRT=0 produces normal output IDMPRT=1 produces detailed 2-phase printout
IPUNCH	Punch flag. IPUNCH=1 punches cards for GEOSYS. Anything else results in no punched output.
IPNPRT	Print flag. IPNPRT=1 prints output sent to card punch. Anything else results in no printout for this section.
DMAX	Maximum molar density anticipated for the system. DMAX=2.0
EPSD	DHFP convergence for densities ($\Delta\rho/\rho$); 0.5E-4
EPSV	FLASH convergence; 0.1E-3
FUGERR	Fugacity convergence; 0.2E-3
EPSS	SEARCH convergence; 0.2E-3
STEP	SEARCH False position initial step size; 1.0002
ENCOR	Correction factor for enthalpies. Needed to produce output consistent with published tables (Btu/lb)
TPHIN	Initial temp. guess for grid calculations (°R) (Use minimum temp. in the grid) Used only with P-H GRID
ISWT(I)	Print flag for ITER loop I. Anything but 0 or blank produces output.
COMP(I,J)	Up to 16 characters for component name

Variable Name	Definition
IDCØM(I)	Sequence I.D. number for component
CMW(I)	Component molecular weight
TC(I)	Component critical temp. (⁰ F)
ACF(I)	Accentric Factor of Component
CD(I)	Component Critical Density (mole/ft ³)
PC(I)	Component Critical pressure (psia)
TB(I)	Component boiling temp. at atmospheric pressure (⁰ F)
CI(I,J)	Ideal Gas Polynomial Coefficients for H and S evaluation (I & EC Process Design and Development, Vol. 11, Page 543)
A(I,J)	Coefficients in modified BWR equation (Fluid Thermodynamic Properties for Light Petroleum Systems, K. E. Starling)
CKIJ(I,J)	Interaction parameters
Z(I)	Component Mole fractions
NUM1	Number of Temperature or Enthalpy grid lines
NUM2	Number of Pressure grid lines
GRID1(I)	Temperature or Enthalpy grid lines (⁰ F or Btu/lb.)
GRID2(I)	Pressure Grid lines (psia)

APPENDIX B

PROGRAM HIERARCHY

APPENDIX C

PROGRAM LISTING AND SAMPLE RUN

```

B      PROGRAM SBWR(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7=PUNSBWR0010
B      *CH)SBWR0020
C      THERMODYNAMIC PROPERTIES TABULATION PROGRAMSBWR0030
A      IMPLICIT REAL*8(A-H,O-Z)SBWR0040
A      DOUBLE PRECISION ISWT,MINX,ICNT,MI,MAXX,ITF,NI
B      REAL ISWT,MINX,ICNT,MI,MAXX,ITF,NI
COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,
1EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE
COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG
COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM
COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG
COMMON/CT/AK(15),AL(15),XL(15),DC,P,R,T,IPRNT,IINM,MD,NC,IPHASE
COMMON/CZ/TB(15),XLMW,CV
COMMON/CY/HIDEAL,SIDEAL
COMMON/CX/VOLM,HMM,SMM,CPMM,CVMM
COMMON/F1/TGRID(30,30),SGRID(30,30),VGRID(30,30),CPGRID(30,30),CVG
*RID(30,30),IPUNCH,NPF,NHF
COMMON/F2/IGRID,IDMPRT
COMMON/F3/GRID1(100),GRID2(100),TPHIN,TGES1,TGES2,NUM1,NUM2,NRUN,IS
*DCOM(15),IPNPRT
COMMON/H/H,S,ENCOR
COMMON/ITERS/ISWT(30)
COMMON/ITERS1/EPS(30),MINX(30),MAXX(30),PCNT1(30),PCNTN(30),
1ICNT(30),ITF(30),XNAM(30),YNAM(30),MI,NI
COMMON/ITERS2/XOLD(30),YOLD(30),XMIN(30),XMAX(30)
DIMENSION HSATV(100),VSATV(100),SSATV(100),HSATL(100),VSATL(100),S
*SATL(100)
DIMENSION TP(100),PT(100),PDP(100),PBP(100),TDP(100),TBP(100)
DIMENSION PH(100),HP(100),HGRID(100)
DIMENSION CPSATV(100),CPSATL(100),CVSATV(100),CVSATL(100)
CALL ECHO
10  FORMAT(8I5)
20  FORMAT(8F10.6)
30  FORMAT(4A4,14,6F10.6)
40  FORMAT(2F9.6,4E13.6,F10.6)
50  FORMAT(6F13.6)
C****
C**** INPUTS
C****
A      5  READ(5,10,END=2000)IGRID,NC,NPHASE,ITNM,IPRNT,IDMPRT,IPUNCH,IPNPRT
B      5  READ(5,10) IGRID,NC,NPHASE,ITNM,IPRNT,IDMPRT,IPUNCH,IPNPRT
IF(EOF(5)) 2000,55,2000
55  READ(5,20) DMAX,EPSS,EPSSV,FUGERR,EPSS,STEP,ENCOR
READ(5,20) TPHIN,(ISWT(I),I=1,6)
DO 110 I=1,NC
READ(5,30) (COMP(J,I),J=1,4),IDCOM(I),CMW(I),TC(I),ACF(I),CD(I),PC
*(I),TB(I)
110 READ(5,40)(CI(I,J),J=1,7)
DO 115 I=1,NC
TC(I)=TC(I)+459.69

```


C-2

```

      ZC(I)=PC(I)/(CD(I)*R*TC(I))
115  TB(I)=TB(I)+459.69
      IF(NC.NE.1) GO TO 130
      READ(5,50) (A(1,J),J=1,NPARM)
      DO 120 J=1,NPARM
120  AL(J)=A(1,J)
      CKIJ(1,1)=0.0
      GO TO 200
130  DO 140 J=1,10
      DO 140 I=1,NC
140  A(I,J)=B(1,J)+B(2,J)*ACF(I)
      DO 150 I=1,NC
A 150  A(I,11)=B(1,11)+B(2,11)*ACF(I)*DEXP(-3.8*ACF(I))
B 150  A(I,11)=B(1,11)+B(2,11)*ACF(I)*EXP(-3.8*ACF(I))
      DO 160 I=1,NC
      CD2 = CD(I)*CD(I)
      TC2 = TC(I)*TC(I)
      TC3 = TC(I)*TC2
      A(I,1)=A(I,1)/CD(I)
      A(I,2)=A(I,2)*R*TC(I)/CD(I)
      A(I,3)=A(I,3)*R*TC3/CD(I)
      A(I,4)=A(I,4)/CD2
      A(I,5)=A(I,5)/CD2
      A(I,6)=A(I,6)*R*TC(I)/CD2
      A(I,7)=A(I,7)/CD(I)**3
      A(I,8)=A(I,8)*R*TC3/CD2
      A(I,9)=A(I,9)*R*TC2*TC2 /CD(I)
      A(I,10)=A(I,10)*R*TC2/CD2
160  A(I,11)=A(I,11)*R*TC3*TC2/CD(I)
      DO 170 I=1,NC
170  CKIJ(1,I)=0.0
      NCM1=NC-1
      DO 180 I=1,NCM1
      IJ=I+1
180  READ(5,20) (CKIJ(1,J),J=IJ,NC)
      DO 190 I=1,NCM1
      IJ=I+1
      DO 190 J=IJ,NC
190  CKIJ(J,I)=CKIJ(I,J)
200  READ(5,20) (Z(I),I=1,NC)
      READ(5,10) NUM1,NUM2
      READ(5,20) (GRID1(I),I=1,NUM1)
      READ(5,20) (GRID2(I),I=1,NUM2)
      NTF=NUM1
      NHF=NUM1
      NPF=NUM2
      DO 210 I=1,NUM1
      TP(I)=GRID1(I)
210  HP(I)=GRID1(I)
      DO 220 I=1,NUM2

```

```

SBWR0510
SBWR0520
SBWR0530
SBWR0540
SBWR0550
SBWR0560
SBWR0570
SBWR0580
SBWR0590
SBWR0600
SBWR0610
SBWR0620
SBWR0630
SBWR0640
SBWR0650
SBWR0660
SBWR0670
SBWR0680
SBWR0690
SBWR0700
SBWR0710
SBWR0720
SBWR0730
SBWR0740
SBWR0750
SBWR0760
SBWR0770
SBWR0780
SBWR0790
SBWR0800
SBWR0810
SBWR0820
SBWR0830
SBWR0840
SBWR0850
SBWR0860
SBWR0870
SBWR0880
SBWR0890
SBWR0900
SBWR0910
SBWR0920
SBWR0930
SBWR0940
SBWR0950
SBWR0960
SBWR0970
SBWR0980
SBWR0990
SBWR1000

```

```

220 PT(I)=GRID2(I)
C**** PH(I)=GRID2(I)
C****
      SUMZ=0.0
      DO 230 I=1,NC
230 SUMZ=SUMZ+Z(I)
      IF(DABS(SUMZ-1.0) .GT. 0.1E-6) WRITE(6,235) SUMZ
      IF( ABS(SUMZ-1.0) .GT. 0.1E-6) WRITE(6,235) SUMZ
235 FORMAT(1H ,///,1H ,33HCOMPOSITIONS DO NOT SUM TO 1.0; SUM = ,G20
      * .8,5X,/,1H ,33HCOMPOSITIONS HAVE BEEN NORMALIZED )
      DO 240 I=1,NC
240 Z(I)=Z(I)/SUMZ
      XLMW=0.0
      DO 250 I=1,NC
250 XLMW=XLMW+XL(I)*CMW(I)
      IF(IGRID .NE. 1) GO TO 270
      DO 260 I=1,NTF
260 TP(I)=TP(I)+459.69
270 CONTINUE
C****
C**** CALL EDIT
C****
C**** WRITE(6,275)
275 FORMAT(1H1)
      IF(IGRID .EQ. 1) WRITE(6,280)
      IF(IGRID .EQ. 0) WRITE(6,281)
280 FORMAT(1H ,28H PRESSURE - TEMPERATURE GRID )
281 FORMAT(1H ,25H PRESSURE - ENTHALPY GRID )
      WRITE(6,283)
283 FORMAT(1H0, 9X,17HTEMPERATURES IN F ,/,10X,17HPRESSURES IN PSIA,
      */,10X,40HSPECIFIC VOLUMES IN CUBIC FEET PER POUND ,/,10X,
      *21HENTHALPIES IN BTU/LB ,/,10X,21HENTROPIES IN BTU/LB-R ,
      */,10X,26HSPECIFIC HEATS IN BTU/LB-R,/,10X,26HM=-1 IMPLIES LIQUID R
      *EGION ,/,10X,24HM=0 IMPLIES VAPOR REGION ,/,10X,
      *26HM=2 IMPLIES 2-PHASE REGION,/)
      WRITE(6,285)
285 FORMAT(1H0,///,1H ,20HSATURATED PROPERTIES )
      WRITE(6,403)
403 FORMAT(///,4X,2HTF ,9X,1HP ,12X,1HV ,9X,1HH ,9X,1HS ,8X,2HCP
      * ,8X,2HCV ,7X,1HM,/)
      IF(IGRID .EQ. 1) GO TO 430
C****
C**** P-H SATURATION PROP. (TSAT GIVEN P)
C****
      PCMAX=PC(1)
      DO 410 I=1,NC

```

```

SBWR1010
SBWR1020
SBWR1030
SBWR1040
SBWR1050
SBWR1060
SBWR1070
SBWR1080
SBWR1090
SBWR1100
SBWR1110
SBWR1120
SBWR1130
SBWR1140
SBWR1150
SBWR1160
SBWR1170
SBWR1180
SBWR1190
SBWR1200
SBWR1210
SBWR1220
SBWR1230
SBWR1240
SBWR1250
SBWR1260
SBWR1270
SBWR1280
SBWR1290
SBWR1300
SBWR1310
SBWR1320
SBWR1330
SBWR1340
SBWR1350
SBWR1360
SBWR1370
SBWR1380
SBWR1390
SBWR1400
SBWR1410
SBWR1420
SBWR1430
SBWR1440
SBWR1450
SBWR1460
SBWR1470
SBWR1480
SBWR1490
SBWR1500

```

```

      IF(PC(I) - PCMAX) 410,410,412
412 PCMAX=PC(I)
410 CONTINUE
      DO 411 I=1,NPF
      P=PH(I)
      IF(P.GT.PCMAX) GO TO 411
      IF(NC.NE.1) GO TO 417
      CALL VATEMP
      TDP(I)=T
      TBP(I)=T
      GO TO 411
417 CALL TDPBPM(TGES1,TGES2,PH(I))
      T=TGES1
407 CALL PDPM
      CALL ITER(2,T,P,PH(I),FLAG)
      IF(FLAG.NE.0.0) GO TO 407
      TDP(I)=T
      T=TGES2
401 CALL PBPM
      CALL ITER(3,T,P,PH(I),FLAG)
      IF(FLAG.NE.0.0) GO TO 401
      TBP(I)=T
      TGES2=T
411 CONTINUE
      GO TO 490
C****
C**** P-T SATURATION PROP. (PSAT GIVEN T)
C****
430 TCMAX=TC(1)
      DO 425 I=1,NC
      IF(TC(I)-TCMAX) 425,425,426
426 TCMAX=TC(I)
425 CONTINUE
      DO 441 I=1,NTF
      T=TP(I)
      IF(T.GT.TCMAX) GO TO 441
      IF(NC.NE.1) GO TO 447
      CALL VAPRES
      PDP(I)=P
      PBP(I)=P
      GO TO 441
447 CALL PDPM
      PDP(I)=P
      CALL PBPM
      PBP(I)=P
      IF(DABS(PDP(I)-PBP(I)).GE.0.01) GO TO 441
      IF(ABS(PDP(I)-PBP(I)).GE.0.01) GO TO 441
      TCMAX=T
441 CONTINUE
490 CONTINUE

```

```

SBWR1510
SBWR1520
SBWR1530
SBWR1540
SBWR1550
SBWR1560
SBWR1570
SBWR1580
SBWR1590
SBWR1600
SBWR1610
SBWR1620
SBWR1630
SBWR1640
SBWR1650
SBWR1660
SBWR1670
SBWR1680
SBWR1690
SBWR1700
SBWR1710
SBWR1720
SBWR1730
SBWR1740
SBWR1750
SBWR1760
SBWR1770
SBWR1780
SBWR1790
SBWR1800
SBWR1810
SBWR1820
SBWR1830
SBWR1840
SBWR1850
SBWR1860
SBWR1870
SBWR1880
SBWR1890
SBWR1900
SBWR1910
SBWR1920
SBWR1930
SBWR1940
SBWR1950
SBWR1960
SBWR1970
SBWR1980
SBWR1990
SBWR2000

```

C****
C**** CALCULATE AND STORE SATURATION PROPERTIES
C****

NDUM=NPF
IF(IGRID .EQ. 1) NDUM=NTF
DO 418 I=1,NDUM
IF(IGRID .EQ. 1) GO TO 470
P=PH(I)
IF(P.GT.PCMAX) GO TO 418
T=TDP(I)
GO TO 475
470 J=JP(I)
IF(T.GT.TCMAX) GO TO 418
P=PDP(I)
475 MD=0
CALL PROPS(O)
HSATV(I)=H
VSATV(I)=V
SSATV(I)=S
CPSATV(I)=CP
CVSATV(I)=CV
IF(IGRID .EQ. 1) GO TO 480
T=TBP(I)
GO TO 485
480 P=PBP(I)
485 MD=-1
CALL PROPS(O)
HSATL(I)=H
VSATL(I)=V
SSATL(I)=S
CPSATL(I)=CP
CVSATL(I)=CV
418 CONTINUE
WRITE(6,420)
420 FORMAT(1H1,15HGRID PROPERTIES)
WRITE(6,403)
IF(IGRID .EQ. 1) GO TO 700

C****
C**** P-H GRID POINT PROPERTIES
C****

DO 600 I=1,NPF
IF(I .EQ. 1) GO TO 604
WRITE(6,603)
603 FORMAT(1H0,20(/))
WRITE(6,403)
604 IDNE=0
WRITE(6,601)
601 FORMAT(/)
TGUESS=TPHIN
DO 600 J=1,NHF

SBWR2010
SBWR2020
SBWR2030
SBWR2040
SBWR2050
SBWR2060
SBWR2070
SBWR2080
SBWR2090
SBWR2100
SBWR2110
SBWR2120
SBWR2130
SBWR2140
SBWR2150
SBWR2160
SBWR2170
SBWR2180
SBWR2190
SBWR2200
SBWR2210
SBWR2220
SBWR2230
SBWR2240
SBWR2250
SBWR2260
SBWR2270
SBWR2280
SBWR2290
SBWR2300
SBWR2310
SBWR2320
SBWR2330
SBWR2340
SBWR2350
SBWR2360
SBWR2370
SBWR2380
SBWR2390
SBWR2400
SBWR2410
SBWR2420
SBWR2430
SBWR2440
SBWR2450
SBWR2460
SBWR2470
SBWR2480
SBWR2490
SBWR2500

```

        IF (PH(I).GT.PCMAX) GO TO 605
        IF ((HP(J).GE.HSATL(I)).AND.(HP(J).LE.HSATV(I))) GO TO 620
        IF (HP(J).GE.HSATV(I)) MD=0.0
        IF (HP(J).LE.HSATL(I)) MD=-1.0
        GO TO 607
605 MD=0.0
607 HGRID(J)=HP(J)
        P=PH(I)
610 T=TGUESS
        CALL PROPS(1)
        HGUESS=H
        CALL ITER(1,TGUESS,HGUESS,HGRID(J),FLAG)
        IF (FLAG.NE.0.0) GO TO 610
        T=TGUESS
        CALL PROPS(0)
        TGRID(I,J)=TGUESS-459.69
        VGRID(I,J)=V
        SGRID(I,J)=S
        CPGRID(I,J)=CP
        CVGRID(I,J)=CV
        GO TO 600
620 IF (NC.EQ. 1) GO TO 623
        IONE=IONE+1
        T=TGUESS
        P=PH(I)
        IF (IONE.NE. 1) GO TO 625
        T=TBP(I)
        CALL ITERM(4,TBP(I),TDP(I))
        CALL VLREGM(1)
        CALL ITER(4,T,HMM,HP(J),FLAG)
        IF (FLAG.NE. 0.0) GO TO 625
        TGUESS=T
        CALL VLREGM(0)
        TGRID(I,J)=TGUESS-459.69
        VGRID(I,J)=VOLM
        SGRID(I,J)=SMM
        CPGRID(I,J)=CPMM
        CVGRID(I,J)=CVMM
        GO TO 600
C****
C**** P-H SINGLE COMPONENT PROP. INSIDE DOME
C****
623 QUAL=(HP(J)-HSATL(I))/(HSATV(I)-HSATL(I))
        TGRID(I,J)=TBP(I)-459.69
        VGRID(I,J)=VSATL(I)+QUAL*(VSATV(I)-VSATL(I))
        SGRID(I,J)=SSATL(I)+QUAL*(SSATV(I)-SSATL(I))
        CPGRID(I,J)=1.03
        CVGRID(I,J)=1.03
        MD=2
        WRITE(6,650) TGRID(I,J),PH(I),VGRID(I,J),HP(J),SGRID(I,J),CPGRID(I,J)

```

```

SBWR2510
SBWR2520
SBWR2530
SBWR2540
SBWR2550
SBWR2560
SBWR2570
SBWR2580
SBWR2590
SBWR2600
SBWR2610
SBWR2620
SBWR2630
SBWR2640
SBWR2650
SBWR2660
SBWR2670
SBWR2680
SBWR2690
SBWR2700
SBWR2710
SBWR2720
SBWR2730
SBWR2740
SBWR2750
SBWR2760
SBWR2770
SBWR2780
SBWR2790
SBWR2800
SBWR2810
SBWR2820
SBWR2830
SBWR2840
SBWR2850
SBWR2860
SBWR2870
SBWR2880
SBWR2890
SBWR2900
SBWR2910
SBWR2920
SBWR2930
SBWR2940
SBWR2950
SBWR2960
SBWR2970
SBWR2980
SBWR2990
SBWR3000

```

650	E,J),CVGRID(I,J),MD	SBWR30010
600	FORMAT(F3.2,3X,F8.3,3X,F9.3,3X,F8.2,3X,F7.4,3X,F7.4,3X,F7.4,3X,12)	SBWR30020
	CONTINUE	SBWR30030
	GO TO 940	SBWR30040
C****		SBWR30050
C****	P-T GRID PROPERTIES	SBWR30060
C****		SBWR30070
700	DO 772 I=1,NIF	SBWR30080
	IF(I.EQ.1) GO TO 704	SBWR30090
	WRITE(6,603)	SBWR30100
	WRITE(6,403)	SBWR30110
704	WRITE(6,601)	SBWR30120
	T=TP(I)	SBWR30130
	IF(T-TCMAX) 782,782,703	SBWR30140
703	PDP(I)=PT(NPF)+10.0	SBWR30150
	PBP(I)=PT(NPF)+10.0	SBWR30160
782	DO 772 J=1,NPF	SBWR30170
	P=PT(J)	SBWR30180
	IF(P-PDP(I)) 776,776,777	SBWR30190
776	MD=C	SBWR30200
	GO TO 778	SBWR30210
777	IF(P-PBP(I)) 779,780,780	SBWR30220
779	CALL VLREGM(G)	SBWR30230
	GO TO 772	SBWR30240
780	MD=-1	SBWR30250
778	CALL PROPS(G)	SBWR30260
772	CONTINUE	SBWR30270
	GO TO 1000	SBWR30280
940	CONTINUE	SBWR30290
C****		SBWR30300
C****	PUNCH OUTPUT	SBWR30310
C****		SBWR30320
	IF(1PNPRT.NE.1) GO TO 1000	SBWR30330
	CALL PUNOUT	SBWR30340
1000	CONTINUE	SBWR30350
	GO TO 5	SBWR30360
2000	CONTINUE	SBWR30370
	STOP	SBWR30380
	END	SBWR30390

A	SUBROUTINE BWRC	BWRC00010
	IMPLICIT REAL*8(A-H,O-Z)	BWRC00020
	COMMON /CB/ A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG	BWRC00030
	COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE	BWRC00040
	IF (NC.NE.1) GO TO 20	BWRC00050
	DO 10 I=1,NPARM	BWRC00060
10	AL(I)=A(1,I)	BWRC00070

	RETURN		BWRC0080
20	THIRD=1./3.		BWRC0090
	DO 30 K=1,NPARM		BWRC0100
30	AL(K)=0.		BWRC0110
	DO 50 I=1,NC		BWRC0120
	AL(1)=AL(1)+XL(I)*A(I,1)		BWRC0130
A	AL(4)=AL(4)+XL(I)*DSQRT(A(I,4))		BWRC0140
B	AL(4)=AL(4)+XL(I)* SORT(A(I,4))		BWRC0150
	DO 40 K=5,8		BWRC0160
40	AL(K)=AL(K)+XL(I)*A(I,K)**THIRD		BWRC0170
50	AL(10)=AL(10)+XL(I)*A(I,10)**THIRD		BWRC0180
	AL(4)=AL(4)**2		BWRC0190
	DO 60 K=5,8		BWRC0200
60	AL(K)=AL(K)**3		BWRC0210
	AL(10)=AL(10)**3		BWRC0220
	DO 70 I=1,NC		BWRC0230
	DO 70 K=1,NC		BWRC0240
	AL(2)=AL(2)+XL(I)*XL(K)*DSQRT(A(I,2)*A(K,2))*(1.-CKIJ(I,K))		BWRC0250
A	AL(2)=AL(2)+XL(I)*XL(K)* SQRT(A(I,2)*A(K,2))*(1.-CKIJ(I,K))		BWRC0260
B	AL(3)=AL(3)+XL(I)*XL(K)*DSQRT(A(I,3)*A(K,3))*(1.-CKIJ(I,K))**3		BWRC0270
A	AL(3)=AL(3)+XL(I)*XL(K)* SQRT(A(I,3)*A(K,3))*(1.-CKIJ(I,K))**3		BWRC0280
B	AL(9)=AL(9)+XL(I)*XL(K)*DSQRT(A(I,9)*A(K,9))*(1.-CKIJ(I,K))**4		BWRC0290
A	AL(9)=AL(9)+XL(I)*XL(K)* SQRT(A(I,9)*A(K,9))*(1.-CKIJ(I,K))**4		BWRC0300
A	70 AL(11)=AL(11)+XL(I)*XL(K)*DSQRT(A(I,11)*A(K,11))*(1.-CKIJ(I,K))**5	BWRC0310	
B	70 AL(11)=AL(11)+XL(I)*XL(K)* SQRT(A(I,11)*A(K,11))*(1.-CKIJ(I,K))**5	BWRC0320	
	RETURN		BWRC0330
	END		BWRC0340

C-8

	SUBROUTINE CVHCAP		CVHC0010
A	IMPLICIT REAL*8(A-H,O-Z)		CVHC0020
	COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,		CVHC0030
1	EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE		CVHC0040
	COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG		CVHC0050
	COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM		CVHC0060
	COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG		CVHC0070
	COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE		CVHC0080
	COMMON/CZ/TB(15),XLMW,CV		CVHC0090
	COMMON/CY/HIDEAL,SIDEAL		CVHC0100
	CV=0.0		CVHC0110
	RETURN		CVHC0120
	END		CVHC0130

	SUBROUTINE DENTE		DENT0010
A	IMPLICIT REAL*8(A-H,O-Z)		DENT0020

```

COMMON /CK/ ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLGD DENT0030
COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE DENT0040
WRITE (6,60) MD DENT0050
IF (IPRNT.NE.0) WRITE (6,70) (AL(I),I=1,11) DENT0060
DELDG=0.03 DENT0070
IF (MD.LT.0) GO TO 10 DENT0080
SGN=1. DENT0090
DC=0. DENT0100
GO TO 20 DENT0110
10 SGN=-1. DENT0120
DC=DMAX DENT0130
20 DO 30 I=1,120 DENT0140
DC=DC+SGN*DELDG DENT0150
CALL PDEVI (FNC) DENT0160
IF (IPRNT.NE.0) WRITE (6,80) I,DC,FNC DENT0170
IF (DABS(FNC/P).LE.EPSD.OR.(DELDG/DC).LE.EPSD) GO TO 50 DENT0180
IF (ABS(FNC/P).LE.EPSD.OR.(DELDG/DC).LE.EPSD) GO TO 50 DENT0190
IF ((SGN*FNC).GT.0.DG) GO TO 40 DENT0200
30 CONTINUE DENT0210
40 DC=DC-SGN*DELDG DENT0220
DELDG=0.382*DELDG DENT0230
GO TO 20 DENT0240
50 IF (DC.GT.0.DG) RETURN DENT0250
DC=P/(R*T) DENT0260
GO TO 20 DENT0270
C DENT0280
60 FORMAT(//,1H,17HCALL TO DENTE(MC=,13,21H):(-1=LIQUID,0=VAPOR) ) DENT0290
70 FORMAT(1H,22H8WRS COEFFICIENTS(AL):,5G17.8,/,6G17.8,/,1H,5HIT DENT0300
* #,11X,2HDC,15X,7HPCALC-P,/) DENT0310
80 FORMAT(1X,15,2G20.8) DENT0320
END DENT0330

```

C-6

```

SUBROUTINE DEWBUB D EWB00010
IMPLICIT REAL*8(A-H,O-Z) D EWB00020
COMMON /CA/ COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS, D EWB00030
1EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE D EWB00040
COMMON /CC/ B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM D EWB00050
COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE D EWB00060
FS=0. D EWB00070
A 10 XV=DLOG(P) D EWB00080
B 10 XV=ALOG(P) D EWB00090
IF (NTYPE.EQ.5) GO TO 40 D EWB00100
20 V=1.0 D EWB00110
DO 30 I=1,NC D EWB00120
30 Y(I)=Z(I) D EWB00130
GO TO 60 D EWB00140
40 V=0. D EWB00150

```


	DD 50 I=1,NC	DEWB0160
	50 X(I)=Z(I)	DEWB0170
	60 CALL SERCH (XV)	DEWB0180
A	P=DEXP(XV)	DEWB0190
B	P= EXP(XV)	DEWB0200
	DD 70 I=1,NC	DEWB0210
	AK(I)=Y(I)/X(I)	DEWB0220
	70 XL(I)=X(I)	DEWB0230
	IF (NC.EQ.1) GO TO 90	DEWB0240
	DD 80 I=1,NC	DEWB0250
A	IF (DABS(AK(I)-1.D0).GT.EPSS) GO TO 100	DEWB0260
B	IF (ABS(AK(I)-1.E0).GT.EPSS) GO TO 100	DEWB0270
	80 CONTINUE	DEWB0280
	WRITE (6,140) NTYPE	DEWB0290
	WRITE(6,320) T,P	DEWB0300
A	90 IF (DABS(DG-DL).LE.EPSS) WRITE (6,150) NTYPE,DG,DL,EPSS	DEWB0310
B	90 IF (ABS(DG-DL).LE.EPSS) WRITE (6,150) NTYPE,DG,DL,EPSS	DEWB0320
	100 MD=-1	DEWB0330
	CALL BWRC	DEWB0340
	DC=DL	DEWB0350
	CALL HSCP	DEWB0360
	HL=HC	DEWB0370
	SL=SC	DEWB0380
	CPL=CP	DEWB0390
	DD 110 I=1,NC	DEWB0400
	110 XL(I)=Y(I)	DEWB0410
	MD=0	DEWB0420
	CALL BWRC	DEWB0430
	DC=DG	DEWB0440
	CALL HSCP	DEWB0450
	HG=HC	DEWB0460
	SG=SC	DEWB0470
	CPG=CP	DEWB0480
	HM=V*HG+(1.E0-V)*HL	DEWB0490
	SM=V*SG+(1.E0-V)*SL	DEWB0500
	CPM=V*CPG+(1.E0-V)*CPL	DEWB0510
	IF (IPRNT.NE.0) WRITE (6,130) T,P,HM,SM	DEWB0520
	120 RETURN	DEWB0530
C	130 FORMAT(//,1H ,6HT(R) = ,G16.8,10H P(PSIA) = ,G16.8,5H HM = ,G16.8,5H SM = ,G16.8)	DEWB0540
	140 FORMAT(//,1H ,33H*****DEWBUB SEARCH FAILED, NTYPE= ,I2,13H;K-VALUE	DEWB0550
	*ES=1.0)	DEWB0560
	150 FORMAT(////,1H ,25H*****DEWBUB SEARCH(NTYPE= ,I2,15H) FAILED: DG	DEWB0570
	* ,G15.8,6H DL = ,G15.8,7H WITHIN ,G15.8)	DEWB0580
	320 FORMAT(10X,14HTEMPERATURE = ,F15.6,10X,11HPRESSURE = ,F15.6)	DEWB0590
	END	DEWB0600
		DEWB0610
		DEWB0620

SUBROUTINE DHFP

DHFP0C10

A	IMPLICIT REAL*8(A-H,O-Z)	
	COMMON /CK/ ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG	DHFPO020
	COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE	DHFPO030
	DIMENSION FN(60), DN(60)	DHFPO040
	NLG=MD	DHFPO050
	KK=0	DHFPO060
10	IT=1	DHFPO070
20	IF (KK.GE.2) GO TO 170	DHFPO080
	IF (NLG.LT.0) GO TO 30	DHFPO090
	DN(IT)=(0.9+0.1*(IT-1))*P/(R*T)	DHFPO100
	GO TO 60	DHFPO110
30	IF (IT.GT.1) GO TO 50	DHFPO120
	SUMT=0.	DHFPO130
	SUMV=0.0	DHFPO140
	DO 40 I=1,NC	DHFPO150
	SUMT=SUMT+XL(I)*TC(I)	DHFPO160
40	SUMV=SUMV+XL(I)/CD(I)	DHFPO170
	TR=I/SUMT	DHFPO180
50	DN(IT)=(1.0-(IT-1)*0.001)/((0.25+0.132*TR)*SUMV)	DHFPO190
60	DC=DN(IT)	DHFPO200
	IF (DC.LT.DMAX.AND.DC.GT.0.E0) GO TO 70	DHFPO210
	FN(IT)=0.	DHFPO220
	IF (NLG) 120,90,90	DHFPO230
70	CALL PDEVI (FNC)	DHFPO240
	FN(IT)=FNC	DHFPO250
	IF (IT.GE.2) GO TO 80	DHFPO260
	IT=2	DHFPO270
	GO TO 20	DHFPO280
80	FN12=FN(IT)-FN(IT-1)	DHFPO290
	FNM=FN(IT)*FN(IT-1)	DHFPO300
A	IF (IT.GE.ITNM.OR.DABS(FN12).LE.1.0-12) GO TO 170	DHFPO310
B	IF (IT.GE.ITNM.OR. ABS(FN12).LE.1.E-12) GO TO 170	DHFPO320
	IF (NLG.LT.0) GO TO 110	DHFPO330
	IF (FN12.LT.0.E0) GO TO 90	DHFPO340
	IF (FNM.GE.0.E0) GO TO 140	DHFPO350
A	IF (DABS(FN(IT)/P).LE.EPSD) GO TO 150	DHFPO360
B	IF (ABS(FN(IT)/P).LE.EPSD) GO TO 150	DHFPO370
90	IF (IPRNT.EQ.C) GO TO 100	DHFPO380
	WRITE (6,200) KK,NLG,(DN(I),I=1,IT)	DHFPO390
	WRITE (6,190) (FN(I),I=1,IT)	DHFPO400
100	NLG=-1	DHFPO410
	KK=KK+1	DHFPO420
	GO TO 10	DHFPO430
110	IF (FN12.GE.0.E0) GO TO 120	DHFPO440
	IF (FNM.GE.0.E0) GO TO 140	DHFPO450
A	IF (DABS(FN(IT)/P).LE.EPSD) GO TO 150	DHFPO460
B	IF (ABS(FN(IT)/P).LE.EPSD) GO TO 150	DHFPO470
120	IF (IPRNT.EQ.0) GO TO 130	DHFPO480
	WRITE (6,200) KK,NLG,(DN(I),I=1,IT)	DHFPO490
	WRITE (6,190) (FN(I),I=1,IT)	DHFPO500
		DHFPO510

130	NLG=0	DHFP0520
	KK=KK+1	DHFP0530
	GO TO 10	DHFP0540
140	IT=IT+1	DHFP0550
	DN(IT)=(DN(IT-2)*FN(IT-1)-DN(IT-1)*FN(IT-2))/FN12	DHFP0560
A	IF (DABS((DN(IT)-DN(IT-1))/DN(IT-1))-EPSD) 160,160,60	DHFP0570
B	IF (ABS((DN(IT)-DN(IT-1))/DN(IT-1))-EPSD) 160,160,60	DHFP0580
150	IT=IT+1	DHFP0590
	DN(IT)=(DN(IT-2)*FN(IT-1)-DN(IT-1)*FN(IT-2))/FN12	DHFP0600
160	DC=DN(IT)	DHFP0610
	IF (IPRNT.EQ.0) RETURN	DHFP0620
	WRITE (6,200) KK,NLG,(DN(I),I=1,IT)	DHFP0630
	ITM=IT-1	DHFP0640
	WRITE (6,190) (FN(I),I=1,ITM)	DHFP0650
	RETURN	DHFP0660
170	IF (IPRNT.EQ.0) GO TO 180	DHFP0670
	WRITE (6,200) KK,NLG,(DN(I),I=1,IT)	DHFP0680
	WRITE (6,190) (FN(I),I=1,IT)	DHFP0690
180	CALL DENTE	DHFP0700
	RETURN	DHFP0710
C		DHFP0720
190	FORMAT(1H ,9HPCALC-P = ,9X,(7G16.8))	DHFP0730
200	FORMAT(1H ,3HKK= ,12,6H; NLG= ,12,5H; DN= , (7G16.8))	DHFP0740
	END	DHFP0750

C-12

A	SUBROUTINE DOMOUT	DOM00010
A	IMPLICIT REAL*8(A-H,O-Z)	DOM00020
B	REAL*8 LM	DOM00030
	REAL LM	DOM00040
	COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,	DOM00050
	EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE	DOM00060
	COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG	DOM00070
	COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM	DOM00080
	COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX ,INLG,NLG	DOM00090
	COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE	DOM00100
	COMMON/CZ/TB(15),XLMW,CV	DOM00110
	COMMON/CX/VOLM,HMM,SMM,CPMM,CVMM	DOM00120
	COMMON/CY/HIDEAL,SIDEAL	DOM00130
	COMMON/F2/IGRID,IDMPRT	DOM00140
	COMMON/H/H,S,ENCOR	DOM00150
	ZMW=0.0	DOM00160
	XMW=0.0	DOM00170
	YMW=0.0	DOM00180
	TF=T-459.69	DOM00190
	IF(IDMPRT .EQ. 0) GO TO 50	DOM00200
	WRITE(6,300)	DOM00210
	IF(IGRID .EQ. 0) GO TO 10.	DOM00220

```

WRITE(6,100) P,TF
GO TO 20
10 WRITE(6,120) P,HMM
20 DO 30 I=1,NC
   ZMW=ZMW+CMW(I)*Z(I)
   XMW=XMW+CMW(I)*X(I)
30 YMW=YMW+CMW(I)*Y(I)
   VM=YMW*V/ZMW
   LM=1.0-VM
   WRITE(6,140) VM,LM
   WRITE(6,160)
   DO 40 I=1,NC
     XM=X(I)*CMW(I)/XMW
     YM=Y(I)*CMW(I)/YMW
40 WRITE(6,180) (COMP(J,I),J=1,4),XM,YM
   WRITE(6,200)
   VLM=1.7/(DL*XMLW)
   HLM=HL/XMW-ENCOR
   HGM=HG/YMW-ENCOR
   SLM=SL/XMW
   SGM=SG/YMW
   CPLM=CPL/XMW
   CPGM=CPG/YMW
   CVLM=0.0
   CVGM=0.0
   VLM=1.7/(DL*XMW)
   VGM=1.7/(DG*YMW)
   WRITE(6,220) TF,P,VLM,HLM,SLM,CPLM,CVLM,MD
   WRITE(6,240) TF,P,VGM,HGM,SGM,CPGM,CVGM,MD
   WRITE(6,260) TF,P,VOLM,HMM,SMM,CPMM,CVMM,MD
   WRITE(6,300)
   GO TO 60
50 WRITE(6,280) TF,P,VOLM,HMM,SMM,CPMM,CVMM,MD
60 RETURN
100 FORMAT(1H0,11HPRESSURE = ,F15.4,10X,14HTEMPERATURE = ,F15.4)
120 FORMAT(1H0,11HPRESSURE = ,F10.3,10X,11HENTHALPY = ,F10.3)
140 FORMAT(1H ,13HWT.% VAPOR = ,F10.5,15X,14HWT.% LIQUID = ,F10.5)
160 FORMAT(1H0,25X,6HLIQUID ,9X,5HVAPOR ,7,26X,6H(WT.%) ,9X,6H(WT.%)
*)
180 FORMAT(1H ,4A4,2F15.5)
200 FORMAT(1H0,24X,2HTF ,9X,1HP ,12X,1HV ,9X,1HH ,9X,1HS ,8X,2HCP
*,8X,2HCV ,7X,1HM )
220 FORMAT(1H ,18HLIQUID PROPERTIES ,2X,F8.2,3X,F8.3,3X,F9.5,3X,F8.2
*,3X,F7.4,3X,F7.4,3X,F7.4,3X,I2)
240 FORMAT(1H ,18HVAPOR PROPERTIES ,2X,F8.2,3X,F8.3,3X,F9.5,3X,F8.2
*,3X,F7.4,3X,F7.4,3X,F7.4,3X,I2)
260 FORMAT(1H ,18HNET PROPERTIES ,2X,F8.2,3X,F8.3,3X,F9.5,3X,F8.2
*,3X,F7.4,3X,F7.4,3X,F7.4,3X,I2)
280 FORMAT(F8.2,3X,F8.3,3X,F9.5,3X,F8.2,3X,F7.4,3X,F7.4,3X,F7.4,3X,I2)
300 FORMAT(1H ,75(1H*))

```

DOM00230
DOM00240
DOM00250
DOM00260
DOM00270
DOM00280
DOM00290
DOM00300
DOM00310
DOM00320
DOM00330
DOM00340
DOM00350
DOM00360
DOM00370
DOM00380
DOM00390
DOM00400
DOM00410
DOM00420
DOM00430
DOM00440
DOM00450
DOM00460
DOM00470
DOM00480
DOM00490
DOM00500
DOM00510
DOM00520
DOM00530
DOM00540
DOM00550
DOM00560
DOM00570
DOM00580
DOM00590
DOM00600
DOM00610
DOM00620
DOM00630
DOM00640
DOM00650
DOM00660
DOM00670
DOM00680
DOM00690
DOM00700
DOM00710
DOM00720

END

DOM00730

SUBROUTINE ECHO
A DIMENSION DATA(20)
B DIMENSION DATA(10)
C

C WRITE HEADER

WRITE(6,99)
WRITE(6,102)
WRITE(6,100)
WRITE(6,101)
WRITE(6,98)
100 FORMAT(1H, 9X,39HOG00000000111111111122222222223333333333 ,
*41H444444444444555555555566666666667777777778)
101 FORMAT(1H, 9X,40H1234567890123456789012345678901234567890 ,
*40H1234567890123456789012345678901234567890)
102 FORMAT(1H,25(1H*),15HINPUT CARD ECHO ,25(1H*),///)
98 FORMAT(1H)
99 FORMAT(1H1)

DATA NIT,NOT/5,6/
ILINE=0

A 300 READ(NIT,200,END=999) DATA

B 300 READ(NIT,200) DATA

A 200 FORMAT(20A4)

B 200 FORMAT(10A8)

B IF(EOF(NIT))999,10,999

10 CONTINUE

ILINE=ILINE+1

WRITE(NOT,11) ILINE,DATA

A 11 FORMAT(1H,I4,5X,20A4)

B 11 FORMAT(1H,I4,5X,10A8)

GO TO 300

999 CONTINUE

A ILINE=ILINE+1

DO 500 I=1,ILINE

500 BACKSPACE NIT

RETURN

END

ECHO00010
ECHO00020
ECHO00030
ECHO00040
ECHO00050
ECHO00060
ECHO00070
ECHO00080
ECHO00090
ECHO00100
ECHO00110
ECHO00120
ECHO00130
ECHO00140
ECHO00150
ECHO00160
ECHO00170
ECHO00180
ECHO00190
ECHO00200
ECHO00210
ECHO00220
ECHO00230
ECHO00240
ECHO00250
ECHO00260
ECHO00270
ECHO00280
ECHO00290
ECHO00300
ECHO00310
ECHO00320
ECHO00330
ECHO00340
ECHO00350
ECHO00360
ECHO00370

C-14

SUBROUTINE EDIT

A IMPLICIT REAL*8(A-H,O-Z)

A REAL*8 ISWT,MINX,ICNT,MI,MAXX,ITF,NI

COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,

EDIT00010
EDIT00020
EDIT00030
EDIT00040

```

1EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE      EDIT00050
COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG                EDIT00060
COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM      EDIT00070
COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG  EDIT00080
COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE  EDIT00090
COMMON/CZ/TB(15),XLMW,CV                                           EDIT00100
COMMON/F1/TGRID(30,30),SGRID(30,30),VGRID(30,30),CPGRID(30,30),CVG EDIT00110
*RID(30,30),IPUNCH,NPF,NHF                                          EDIT00120
COMMON/F2/IGRID,IDMPRT                                              EDIT00130
COMMON/F3/GRID1(100),GRID2(100),TPHIN,TGES1,TGES2,NUM1,NUM2,NRUN,I EDIT00140
*DCOM(15),IPNPRT                                                    EDIT00150
COMMON/H/H,S,ENCOR                                                 EDIT00160
COMMON/ITERS/ISWT(30)                                              EDIT00170
WRITE(6,20)                                                         EDIT00180
WRITE(6,60) IGRID,NC,NPHASE,ITNM,IPRNT,IDMPRT,IPUNCH,IPNPRT      EDIT00190
WRITE(6,80) DMAX,EPSD,EPSV,FUGERR,EPSS,STEP,ENCOR                EDIT00200
WRITE(6,100) TPHIN,(ISWT(I),I=1,6)                                EDIT00210
WRITE(6,110)                                                         EDIT00220
DO 500 I=1,NC                                                       EDIT00230
WRITE(6,120) (COMP(J,I),J=1,4),IDCOM(I),CMW(I),TC(I),ACF(I),CD(I), EDIT00240
PC(I),TB(I)                                                         EDIT00250
500 WRITE(6,140) (CI(I,J),J=1,7)                                   EDIT00260
WRITE(6,160) ((A(I,J),J=1,NPARM),I=1,NC)                          EDIT00270
WRITE(6,170)                                                         EDIT00280
DO 510 I=1,NC                                                       EDIT00290
510 WRITE(6,180) (CKIJ(I,J),J=1,NC)                                EDIT00300
WRITE(6,190)                                                         EDIT00310
DO 520 I=1,NC                                                       EDIT00320
520 WRITE(6,200) (COMP(J,I),J=1,4),Z(I)                             EDIT00330
WRITE(6,220) NUM1,NUM2                                              EDIT00340
WRITE(6,240) (GRID1(I),I=1,NUM1)                                   EDIT00350
WRITE(6,260) (GRID2(I),I=1,NUM2)                                   EDIT00360
C****                                                              EDIT00370
C****                                                              EDIT00380
20 FORMAT(1H1,25(1H*),14HEDIT THIS CASE ,25(1H*),///)           EDIT00390
60 FORMAT(1H0,47HIGRID,NC,NPHASE,ITNM,IPRNT,IDMPRT,IPUNCH,IPNPRT , EDIT00400
*,1X,8I5)                                                           EDIT00410
80 FORMAT(1H0,37HDMAX,EPSD,EPSV,FUGERR,EPSS,STEP,ENCOR ,/,1X,7E15.6) EDIT00420
100 FORMAT(1H0,21HTPHIN,(ISWT(I),I=1,6) ,/,1X,7F15.5)           EDIT00430
110 FORMAT(1H0,64H(COMP(J,I),J=1,4),IDCOM(I),CMW(I),TC(I),ACF(I),CD(I) EDIT00440
*,PC(I),TB(I) )                                                    EDIT00450
120 FORMAT(1H0,4A4,14,6F10.4)                                       EDIT00460
140 FORMAT(1H ,2HC1 ,7E15.6)                                         EDIT00470
160 FORMAT(1H0,27H((A(I,J),J=1,NPARM),I=1,NC) ,/,(1X,7E15.6))   EDIT00480
170 FORMAT(1H0,10HCKIJ(I,J) )                                       EDIT00490
180 FORMAT(1H ,7E15.5)                                              EDIT00500
190 FORMAT(1H0,24HCOMPONENT MOLE FRACTIONS )                       EDIT00510
200 FORMAT(1H ,4A4,F15.5)                                           EDIT00520
220 FORMAT(1H0,13HNUM1 AND NUM2 ,5X,2I5)                             EDIT00530
240 FORMAT(1H0,34HTEMPERATURE OR ENTHALPY GRID LINES ,/,(1X,10F10.2)) EDIT00540

```

```

260 FORMAT(1HC,19HPRESSURE GRID LINES ,/, (1X,10F10.2))
      RETURN
      END

```

```

EDIT0550
EDIT0560
EDIT0570

```

```

A      SUBROUTINE F
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CA/ COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,
1     EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE
      COMMON /CB/ A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG
      COMMON /CK/ ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG
      COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE
      IF (MD.LT.0) GO TO 20
      DO 10 I=1,NC
10     XL(I)=Y(I)
      GO TO 40
      DO 30 I=1,NC
      DO 30 I=1,NC
30     XL(I)=X(I)
40     CALL BWRC
      CALL DHFP
      CALL FUGA
      IF (MD.LT.0) GO TO 60
      DG=DC
      IF (NLG.NE.MD) INLG=0
      DO 50 I=1,NC
50     FUG(I)=FU(I)
      RETURN
60     DL=DC
      IF (NLG.NE.MD) INLG=-1
      DO 70 I=1,NC
70     FUL(I)=FU(I)
      RETURN
      END

```

```

FFFF0010
FFFF0020
FFFF0030
FFFF0040
FFFF0050
FFFF0060
FFFF0070
FFFF0080
FFFF0090
FFFF0100
FFFF0110
FFFF0120
FFFF0130
FFFF0140
FFFF0150
FFFF0160
FFFF0170
FFFF0180
FFFF0190
FFFF0200
FFFF0210
FFFF0220
FFFF0230
FFFF0240
FFFF0250
FFFF0260
FFFF0270
FFFF0280

```

C-16

```

A      SUBROUTINE FLASH
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CA/ COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,
1     EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE
      COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE
      TESTV=0.0
      TESTL=0.0
      DO 10 N=1,NC
      TESTL=TESTL+AK(N)*Z(N)
10     TESTV=TESTV+Z(N)/AK(N)
      IF (IPRNT.NE.0) WRITE (6,120) TESTL,TESTV

```

```

FLAS0010
FLAS0020
FLAS0030
FLAS0040
FLAS0050
FLAS0060
FLAS0070
FLAS0080
FLAS0090
FLAS0100
FLAS0110

```

	IF (TESTL.GT.1.E0) GO TO 20	FLAS0120
	V=0.	FLAS0130
	GO TO 70	FLAS0140
20	IF (TESTV.GT.1.0E0) GO TO 30	FLAS0150
	V=1.0	FLAS0160
	GO TO 70	FLAS0170
30	DO 50 IT=1,ITNM	FLAS0180
	FVA=0.0	FLAS0190
	FVAD=0.0	FLAS0200
	DO 40 N=1,NC	FLAS0210
	FVAN=(Z(N)*(AK(N)-1.E0))/(V*(AK(N)-1.E0)+1.E0)	FLAS0220
	FVA=FVA+FVAN	FLAS0230
40	FVAD=FVAD+FVAN*FVAN/Z(N)	FLAS0240
	V=V+FVA/FVAD	FLAS0250
	IF (IPRNT.NE.0) WRITE (6,110) V	FLAS0260
A	IF (DABS(FVA).LT.EPSV) GO TO 60	FLAS0270
B	IF (ABS(FVA).LT.EPSV) GO TO 60	FLAS0280
	IF (V.GT.1.E0) V=(V-FVA/FVAD+1.E0)*5.E-1	FLAS0290
	IF (V.LT.0.E0) V=(V-FVA/FVAD)*5.E-1	FLAS0300
50	CONTINUE	FLAS0310
	WRITE (6,100) ITNM,V	FLAS0320
60	IF (V.LT.0.E0) V=0.E0	FLAS0330
	IF (V.GT.1.E0) V=1.E0	FLAS0340
70	SUMX=0.0	FLAS0350
	SUMY=0.	FLAS0360
	DO 80 N=1,NC	FLAS0370
	Y(N)=(Z(N)*AK(N))/(V*(AK(N)-1.E0)+1.E0)	FLAS0380
	X(N)=Y(N)/AK(N)	FLAS0390
	SUMX=SUMX+X(N)	FLAS0400
80	SUMY=SUMY+Y(N)	FLAS0410
	DO 90 N=1,NC	FLAS0420
	X(N)=X(N)/SUMX	FLAS0430
90	Y(N)=Y(N)/SUMY	FLAS0440
	RETURN	FLAS0450
C	100 FORMAT(///,1H ,25HFLASH FAILED TO CONVERGE: ,I5,1CH ITS; V = ,G	FLAS0460
	*15.8,7)	FLAS0470
	110 FORMAT(1H ,1HV ,G15.8)	FLAS0480
	120 FORMAT(1H ,11HTESTL,TESTV ,2G15.8)	FLAS0490
	END	FLAS0500
		FLAS0510

A	SUBROUTINE FUGA	FUGA0010
	IMPLICIT REAL*8(A-H,D-Z)	FUGA0020
	COMMON /CB/ A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG	FUGA0030
	COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE	FUGA0040
	THIRD=1./3.	FUGA0050
	DC3=DC**3	FUGA0060

		BLE=AL(4)*DC*DC	FUGA0070
A		EBLE=DEXP(-BLE)	FUGA0080
B		EBLE= EXP(-BLE)	FUGA0090
		DO 20 K=1,NC	FUGA0100
		FU11=0.	FUGA0110
		FU12=0.	FUGA0120
		FU13=0.	FUGA0130
		FU14=0.	FUGA0140
		DO 10 I=1,NC	FUGA0150
A		FU11=FU11+XL(I)*DSQRT(A(K,2)*A(I,2))*(1.0-CKIJ(K,I))	FUGA0160
B		FU11=FU11+XL(I)* SQRT(A(K,2)*A(I,2))*(1.0-CKIJ(K,I))	FUGA0170
A		FU12=FU12+XL(I)*DSQRT(A(K,3)*A(I,3))*(1.0-CKIJ(K,I))**3	FUGA0180
B		FU12=FU12+XL(I)* SQRT(A(K,3)*A(I,3))*(1.0-CKIJ(K,I))**3	FUGA0190
A		FU13=FU13+XL(I)*DSQRT(A(K,9)*A(I,9))*(1.0-CKIJ(K,I))**4	FUGA0200
B		FU13=FU13+XL(I)* SQRT(A(K,9)*A(I,9))*(1.0-CKIJ(K,I))**4	FUGA0210
A	10	FU14=FU14+XL(I)*DSQRT(A(K,11)*A(I,11))*(1.0-CKIJ(K,I))**5	FUGA0220
B	10	FU14=FU14+XL(I)* SQRT(A(K,11)*A(I,11))*(1.0-CKIJ(K,I))**5	FUGA0230
A		FU15=DSQRT(A(K,4)/AL(4))	FUGA0240
B		FU15= SQRT(A(K,4)/AL(4))	FUGA0250
		FU21=(AL(5)**2*A(K,5))**THIRD	FUGA0260
		FU22=(AL(6)**2*A(K,6))**THIRD	FUGA0270
		FU23=(AL(10)**2*A(K,10))**THIRD	FUGA0280
		FU24=(AL(7)**2*A(K,7))**THIRD	FUGA0290
		FU25=(AL(8)**2*A(K,8))**THIRD	FUGA0300
		B1=R*(A(K,1)+AL(1)+1.5*DC*FU21)	FUGA0310
		B2=-2.*FU11+DC*(-1.5*FU22+0.6*DC3*(AL(7)*FU22+AL(6)*FU24))	FUGA0320
		B3=DC*(-1.5*FU23+0.6*DC3*(AL(7)*FU23+AL(10)*FU24))	FUGA0330
		B4=-2.*FU12+3.*DC*FU25*((1.-EBLE)/BLE-0.5*EBLE)-2.*DC*AL(8)/BLE*FU	FUGA0340
		115*(1.-EBLE*(1.+BLE+0.5*BLE*BLE))	FUGA0350
		SFU=((((B1*T+B2)*T+B3)*T+B4)*T+2.*FU13)*T-2.*FU14)*DC/(R*T**5)	FUGA0360
A	20	FU(K)=DC*R*T*XL(K)*DEXP(SFU)	FUGA0370
B	20	FU(K)=DC*R*T*XL(K)* EXP(SFU)	FUGA0380
		RETURN	FUGA0390
		END	FUGA0400

C-18

		SUBROUTINE FUNC (XV,YV)	FUNC0010
A		IMPLICIT REAL*8(A-H,O-Z)	FUNC0020
		COMMON /CA/ COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,	FUNC0030
		1EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE	FUNC0040
		COMMON /CC/ B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM	FUNC0050
		COMMON /CK/ ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG	FUNC0060
		COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE	FUNC0070
		INLG=1	FUNC0080
A	10	P=DEXP(XV)	FUNC0090
B	10	P= EXP(XV)	FUNC0100
		IF(NTYPE .EQ. 5) GO TO 60	FUNC0110
	30	MD=0	FUNC0120

```

      CALL F
      SUMX=0.
      DO 40 I=1,NC
      X(I)=Y(I)/AK(I)
40    SUMX=SUMX+X(I)
      DO 50 I=1,NC
50    X(I)=X(I)/SUMX
      MD=-1
      CALL F
      GO TO 90
60    MD=-1
      CALL F
      SUMY=0.
      DO 70 I=1,NC
      Y(I)=X(I)*AK(I)
70    SUMY=SUMY+Y(I)
      DO 80 I=1,NC
80    Y(I)=Y(I)/SUMY
      MD=0
      CALL F
      DO 100 I=1,NC
100   AK(I)=FUL(I)*Y(I)/(FUG(I)*X(I))
      IF (IPRNT.EQ.0) GO TO 120
      WRITE (6,250)
      DO 110 I=1,NC
110   WRITE (6,260) (COMP(J,I),J=1,4),Y(I),X(I),AK(I),FUG(I),FUL(I)
120   SUM=0.0
      IF (NTYPE .EQ. 5) GO TO 140
      DO 130 I=1,NC
130   SUM=SUM+Y(I)/AK(I)
      GO TO 160
140   DO 150 I=1,NC
150   SUM=SUM+X(I)*AK(I)
160   YV=SUM-1.
220   IF (IPRNT.NE.0) WRITE (6,240) YV,INLG,T,P
230   RETURN
C
240   FORMAT(1H ,16HSEARCH VARIABLE= ,G16.8,8H INLG = ,I2,/,7H T(R) =
*      ,G16.8,9HP(PSIA) = ,G16.8)
250   FORMAT(24X,1HY ,14X,1HX ,14X,1HK ,14X,3HFUG ,12X,3HFUL )
260   FORMAT (1X,4A4,5G15.8)
      END

```

```

FUNC0130
FUNC0140
FUNC0150
FUNC0160
FUNC0170
FUNC0180
FUNC0190
FUNC0200
FUNC0210
FUNC0220
FUNC0230
FUNC0240
FUNC0250
FUNC0260
FUNC0270
FUNC0280
FUNC0290
FUNC0300
FUNC0310
FUNC0320
FUNC0330
FUNC0340
FUNC0350
FUNC0360
FUNC0370
FUNC0380
FUNC0390
FUNC0400
FUNC0410
FUNC0420
FUNC0430
FUNC0440
FUNC0450
FUNC0460
FUNC0470
FUNC0480
FUNC0490
FUNC0500
FUNC0510
FUNC0520
FUNC0530
FUNC0540

```

```

A  SUBROUTINE HSCP
    IMPLICIT REAL*8(A-H,O-Z)
    COMMON /CC/ R(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM
    COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE

```

```

HSCP0010
HSCP0020
HSCP0030
HSCP0040

```

```

COMMON/CY/HIDEAL,SIDEAL
T5=T**5
DC2=DC*DC
DC3=DC2*DC
BLE=AL(4)*DC2
BLE2=BLE*BLE
A EBLE=DEXP(-BLE)
B EBLE=EXP(-BLE)
B1=(AL(1)+AL(5)*DC)*R
B2=(1.2*AL(7)*AL(6)*DC3-1.5*AL(6))*DC-2.*AL(2)
B3=2.*AL(10)*DC*(0.7*AL(7)*DC**3-1.)
B4=(3.-(3.+BLE*0.5-BLE*BLE)*EBLE)*AL(8)/BLE*DC-4.*AL(3)
HC=(((((B1*T+B2)*T+B3)*T+B4)*T+5.*AL(9))*T-6.*AL(11))*DC/T**4*CONV
HIDEAL=0.
DO 10 I=1,NC
10 HIDEAL=HIDEAL+CMW(I)*((((CI(I,6)*T+CI(I,5))*T+CI(I,4))*T+CI(I,3))
1*T+CI(I,2))*T+CI(I,1))*XL(I)
HC=HC+HIDEAL
B1=-R*(AL(1)+AL(5)*DC*0.5)
B2=DC*AL(10)*(0.2*AL(7)*DC3-0.5)
B3=2.*(AL(8)*DC/BLE*(1.-(1.+0.5*BLE)*EBLE)-AL(3))
A SC=(((((B1*T+T+B2)*T+B3)*T+3.*AL(9))*T-4.*AL(11))*DC/T5-R*DLOG(R*T*
B SC=(((((B1*T+T+B2)*T+B3)*T+3.*AL(9))*T-4.*AL(11))*DC/T5-R*ALOG(R*T*
1DC*0.068046)
SIDEAL=0.
XLNX=0.
DO 20 I=1,NC
SIDEAL=SIDEAL+CMW(I)*XL(I)*((((1.25*CI(I,6)*T+4./3.*CI(I,5))*T+1.5
A 1*CI(I,4))*T+2.*CI(I,3))*T+CI(I,2)*DLOG(T)+CI(I,7))
B 1*CI(I,4))*T+2.*CI(I,3))*T+CI(I,2)*ALOG(T)+CI(I,7))
A 20 XLNX=XLNX+XL(I)*DLOG(XL(I))
B 20 XLNX=XLNX+XL(I)*ALOG(XL(I))
SC=(SC-R*XLNX)*CONV+SIDEAL
DPDTD=DC*R*(1.+(AL(1)+AL(5)*DC)*DC)+(((AL(10)*DC*(1.-AL(7)*DC3)*T+
12.*(AL(3)-AL(8)*DC*(1.+BLE)*EBLE))*T-3.*AL(9))*T+4.*AL(11))*DC2/T5
DPDDT=R*T*(1+DC*(2.*AL(1)+3.*AL(5)*DC))+DC*((6.*AL(7)*AL(6)*DC3-3.
1*AL(6))*DC-2.*AL(2)+((3.*AL(10)*DC*(2.*AL(7)*DC3-1.)*T+AL(8)*DC*(
23.+3.*BLE-2.*BLE2)*EBLE-2.*AL(3))*T+2.*AL(9))*T-2.*AL(11))/T**4)
DSDTD=-R/T+(((AL(10)*DC*(1.-4*AL(7)*DC3)*T+6.*(AL(3)-AL(8)*DC/BLE
1*(1.-(1.+0.5*BLE)*EBLE))*T-12.*AL(9))*T+20.*AL(11))*DC/(T5*T)
CPD=(DSDTD+DPDTD**2/(DC2*DPDDT))*T*CONV
CPI=0.
DO 30 I=1,NC
30 CPI=CMW(I)*XL(I)*((((5*CI(I,6)*T+4*CI(I,5))*T+3*CI(I,4))*T+2*CI(I,
13))*T+CI(I,2))+CPI
CP=CPD+CPI
RETURN
END

```

HSCPO050
HSCPO060
HSCPO070
HSCPO080
HSCPO090
HSCPO100
HSCPO110
HSCPO120
HSCPO130
HSCPO140
HSCPO150
HSCPO160
HSCPO170
HSCPO180
HSCPO190
HSCPO200
HSCPO210
HSCPO220
HSCPO230
HSCPO240
HSCPO250
HSCPO260
HSCPO270
HSCPO280
HSCPO290
HSCPO300
HSCPO310
HSCPO320
HSCPO330
HSCPO340
HSCPO350
HSCPO360
HSCPO370
HSCPO380
HSCPO390
HSCPO400
HSCPO410
HSCPO420
HSCPO430
HSCPO440
HSCPO450
HSCPO460
HSCPO470
HSCPO480
HSCPO490
HSCPO500
HSCPO510
HSCPO520

```

A      IMPLICIT REAL*8 (A-Z)
B      IMPLICIT REAL (A-Z)
        INTEGER I,J,ICLK
        COMMON/ITERS/ISWT(30)
        COMMON/ITERS1/EPST(30),MINX(30),MAXX(30),PCNT1(30),PCNTN(30),
        1ICNT(30),ITF(30),XNAM(30),YNAM(30),MI,NI
        COMMON/ITERS2/XOLD(30),YOLD(30),XMIN(30),XMAX(30)
C      GENERAL NEWTON TYPE ITERATION SCHEME
C      CALL ITER(I,X,YI,YJ,*)
C      I - ITERATION NUMBER
C      X - INDEP VARIABLE
C      YI - DEP VARIABLE F(X)
C      YJ - DEP VARIABLE G(X)
C      ISWT - PRINT FLAG
C      DEL - DEP VARIABLE TOLERANCE
C      MINX - MINIMUM X
C      MAXX - MAXIMUM X
C      PCNT1 - FRACTIONAL AMOUNT TO INCREMENT 1ST X
C      PCNTN - FRACTIONAL AMOUNT TO INCREMENT NTH X
C      ICNT - COUNTER(1-15,NOT CLOSED) (0,CLOSED)
C      ITF - FLAG, POS=NEWTONS, NEG=COUNTER TYPE ITERATION
C      XNAM - INDEP VARIABLE NAME(8 CHAR MAX)
C      MIJQ - NON-CLOSURE FLAG --- NON-CLOSURE = NON-ZERO
C      YNAM - DEP VARIABLE NAME(8 CHAR MAX)
C
C      IF(ICNT(I).NE.0.) GO TO 10
C      SAVE ORIGINAL MIN,MAX X ON 1ST PASS
C      XMIN(I) = MINX(I)
C      XMAX(I) = MAXX(I)
C      10 DIFY = YI-YJ
C      EPSY = DABS(DIFY/YJ)
C      EPSY = ABS(DIFY/YJ)
C      CLOSURE TOLERANCE TEST
C      IF(EPSY.GT.EPS(1))GO TO 20
C      ICNT(I) = 0.
C      GO TO 80
C      CHECK FOR COUNTER ONLY 7ATH
C      20 IF(ICNT(I).GE.NI) GO TO 160
C      IF(ITF(I).LT.0.0) GO TO 70
C      IF(ICNT(I).NE.0.)GO TO 30
C      1ST PASS X INCREMENT CALCULATION
C      DELX = DSIGN(PCNT1(I),-DIFY) * DABS(X)
C      DELX = SIGN(PCNT1(I),-DIFY) * ABS(X)
C      GO TO 40
C      30 DIFX = X -XOLD(I)
C      DELY=YOLD(I)-DIFY
C      NTH PASS X INCREMENT CALCULATION
C      IF(DELY.EQ.0.0) GO TO 36
C      DELX= PCNTN(I)*DIFX*DIFY/DELY
C      Z= X+DELX

```

```

ITER0020
ITER0030
ITER0040
ITER0050
ITER0060
ITER0070
ITER0080
ITER0090
ITER0100
ITER0110
ITER0120
ITER0130
ITER0140
ITER0150
ITER0160
ITER0170
ITER0180
ITER0190
ITER0200
ITER0210
ITER0220
ITER0230
ITER0240
ITER0250
ITER0260
ITER0270
ITER0280
ITER0290
ITER0300
ITER0310
ITER0320
ITER0330
ITER0340
ITER0350
ITER0360
ITER0370
ITER0380
ITER0390
ITER0400
ITER0410
ITER0420
ITER0430
ITER0440
ITER0450
ITER0460
ITER0470
ITER0480
ITER0490
ITER0500
ITER0510

```

	IF(X.GT.Z.AND.YI.GT.YJ) GO TO 40	ITER0520
	IF(X.LT.Z.AND.YI.LT.YJ) GO TO 40	ITER0530
	IF(X.GT.Z.AND.YI.LT.YJ) GO TO 36	ITER0540
	IF(X.GE.Z. OR.YI.LE.YJ) GO TO 40	ITER0550
A	36 DELX=DABS(X)*DSIGN(5.D-2,-DIFY)	ITER0560
B	36 DELX= ABS(X)* SIGN(5.E-2,-DIFY)	ITER0570
	40 XOLD(I) = X	ITER0580
	IF(DELX) 42,45,41	ITER0590
	41 MINX(I)= XOLD(I)	ITER0600
	GO TO 43	ITER0610
	42 MAXX(I)= XOLD(I)	ITER0620
	43 X RANGE= (MAXX(I)-MINX(I))*0.5	ITER0630
A	IF(DABS(DELX).LT.XRANGE) GO TO 45	ITER0640
B	IF(ABS(DELX).LT.XRANGE) GO TO 45	ITER0650
	IF(MAXX(I).EQ.XMAX(I)) GO TO 44	ITER0660
	IF(MINX(I).NE.XMIN(I)) GO TO 45	ITER0670
A	44 DELX=DSIGN(XRANGE,DELX)	ITER0680
B	44 DELX= SIGN(XRANGE,DELX)	ITER0690
A	45 IF(DELX.EQ.0.0) DELX=DSIGN(1.E-2,-DIFY)	ITER0700
B	45 IF(DELX.EQ.0.0) DELX= SIGN(1.E-2,-DIFY)	ITER0710
C	INCREMENT X	ITER0720
	X = X + DELX	ITER0730
	IF(MINX(I).LI.X) GO TO 50	ITER0740
C	X EXCEEDS MIN, X = MIN	ITER0750
	X = MINX(I)	ITER0760
	GO TO 60	ITER0770
C	X EXCEEDS MAX, X = MAX	ITER0780
	50 IF(X.GT.MAXX(I))X = MAXX(I)	ITER0790
	60 IF(X.NE.XOLD(I))GO TO 70	ITER0800
	IF(ICNT(I).NE. 0.)GO TO 160	ITER0810
	70 ICNT(I) = ICNT(I) + 1.	ITER0820
C	PRINT AFTER THE 10TH ITERATION	ITER0830
	IF(ICNT(I).LE.MI)GO TO 80	ITER0840
C	FORCE CLOSURE AFTER 15 PASSES	ITER0850
	IF(ICNT(I).GT.MI)GO TO 160	ITER0860
	GO TO 90	ITER0870
C	PRINT EVERY PASS IF FLAGGED	ITER0880
	80 IF(ISWT(I).EQ. 0.)GO TO 100	ITER0890
	90 CONTINUE	ITER0900
	95 WRITE(6,180)I,ICNT(I),XNAM(I),XOLD(I),X,YNAM(I),YI,YJ,EPSY	ITER0910
	180 FORMAT(1H,1HL,13,2H N, F3.0,1X,A8,G12.5,1X,G12.5,1X,A8,G12.5,1	ITER0920
	*X,G12.5,1X,2HE,1PE9.3)	ITER0930
	100 IF(ICNT(I).EQ. 0.)GO TO 170	ITER0940
	IF(ITF(I).LT.0.)GO TO 140	ITER0950
	YOLD(I) = DIFY	ITER0960
C	NON-CLOSURE RETURN	ITER0970
	140 GO TO 175	ITER0980
A	ENTRY ITERZ(I)	ITER0990
B	ENTRY ITERZ	ITER1000
	160 IF(ICNT(I).EQ.0.)GO TO 175	ITER1010

```

      ICNT(I) = 0.
      GO TO 90
C 170 RESTORE ORIGINAL MIN,MAX X ON NTH PASS
      MINX(I) = XMIN(I)
      MAXX(I) = XMAX(I)
C 175 NORMAL RETURN(CLOSURE)
      FLAG= ICNT(I)
      RETURN
      END
      SUBROUTINE ITEM(J,MIN,MAX)
C 176 SUBROUTINE TO REINITIALIZE TOLERANCE, MINIMUM, AND MAXIMUM
A 177 IMPLICIT REAL*8 (A-Z)
B 178 IMPLICIT REAL (A-Z)
      INTEGER J
      COMMON/ITERS1/EPSS(30),MINX(30),MAXX(30),PCNT1(30),PCNTN(30),
1 ICNT(30),ITF(30),XNAM(30),YNAM(30),MI,NI
      MINX(J)=MIN
      MAXX(J)=MAX
      RETURN
      END

```

```

ITER1020
ITER1030
ITER1040
ITER1050
ITER1060
ITER1070
ITER1080
ITER1090
ITER1100
ITEM0010
ITEM0020
ITEM0030
ITEM0040
ITEM0050
ITEM0060
ITEM0070
ITEM0080
ITEM0090
ITEM0100
ITEM0110

```

```

      SUBROUTINE KI
A 179 IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CA/ COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,
1 EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE
      COMMON /CK/ ACF(15),CO(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG
      COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE
      IF (NC.NE.1) GO TO 10
      AK(1)=1.0
      RETURN
10 DO 20 I=1,NC
      TR=T/TC(I)
A 20 AK(I)=PC(I)/P*DEXP(2.3026*((16.25-73.85*ZC(I)+90.*ZC(I)**2)*(TR-1.
B 20 AK(I)=PC(I)/P* EXP(2.3026*((16.25-73.85*ZC(I)+90.*ZC(I)**2)*(TR-1.
1)/TR-10.**(-8.68*(TR-1.8+6.2*ZC(I))**2)))
      TESTV=0.0
      DO 30 I=1,NC
      TESTV=TESTV+Z(I)/AK(I)
30 IF (TESTV.GT.1.E0) GO TO 50
      DO 40 I=1,NC
40 AK(I)=AK(I)*TESTV*0.99999999E0
50 TESTL=0.0
      DO 60 I=1,NC
60 TESTL=TESTL+AK(I)*Z(I)
      IF (TESTL.GT.1.E0) GO TO 80
      DO 70 I=1,NC
70 AK(I)=AK(I)/TESTL*1.0000000100

```

```

KII10010
KII10020
KII10030
KII10040
KII10050
KII10060
KII10070
KII10080
KII10090
KII10100
KII10110
KII10120
KII10130
KII10140
KII10150
KII10160
KII10170
KII10180
KII10190
KII10200
KII10210
KII10220
KII10230
KII10240
KII10250
KII10260

```

```

80 IF (IPRNT.NE.0) WRITE (6,90) (AK(I),I=1,NC)
RETURN
C
90 FORMAT(////,1H ,3HAKI ,8G15.8)
END

```

```

KIIIO270
KIIIO280
KIIIO290
KIIIO300
KIIIO310

```

```

A SUBROUTINE PBPM
  IMPLICIT REAL*8(A-H,O-Z)
  COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,
1 EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE
  COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG
  COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM
  COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG
  COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE
  COMMON/XYZ/PDPBPI
  NTYPE=5
  P=1.0
  DO 2 I=1,NC
  TR=TC(I)
A 2 AK(I)=PC(I)/P*DEXP(2.3026*((16.25-73.85*ZC(I)+90.*ZC(I)**2)*(TR-1.
B 2 AK(I)=PC(I)/P* EXP(2.3026*((16.25-73.85*ZC(I)+90.*ZC(I)**2)*(TR-1.
1)/TR-10.**(-8.68*(TR-1.8+6.2*ZC(I))**2)))
  CALL PDPBPM
  P=PDPBPI
  PF=P
  IFLAG=1
  CALL DEWBUB
  RETURN
END

```

```

PBPM0010
PBPM0020
PBPM0030
PBPM0040
PBPM0050
PBPM0060
PBPM0070
PBPM0080
PBPM0090
PBPM0100
PBPM0110
PBPM0120
PBPM0130
PBPM0140
PBPM0150
PBPM0160
PBPM0170
PBPM0180
PBPM0190
PBPM0200
PBPM0210
PBPM0220
PBPM0230

```

```

A SUBROUTINE PDEVI (FNC)
  IMPLICIT REAL*8(A-H,O-Z)
  COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE
  D2=DC*DC
  BLE=AL(4)*D2
  B1=R*(1./DC+AL(1)+AL(5)*DC)
  B2=AL(7)*AL(6)*D2*D2-AL(6)*DC-AL(2)
  B3=AL(10)*DC*(AL(7)*D2*DC-1.)
  A B4=AL(8)*DC*(1.+BLE)*DEXP(-BLE)-AL(3)
  B4=AL(8)*DC*(1.+BLE)* EXP(-BLE)-AL(3)
  FNC=((((B1*T+B2)*T+B3)*T+B4)*T+AL(9))*T-AL(11))*D2/T**4-P
  RETURN
END

```

```

PDEVOC10
PDEVOC20
PDEVOC30
PDEVOC40
PDEVOC50
PDEVOC60
PDEVOC70
PDEVOC80
PDEVOC90
PDEVOC100
PDEVOC110
PDEVOC120
PDEVOC130

```

SUBROUTINE PDPM

PDPM0010

A		IMPLICIT REAL*8(A-H,O-Z)	PDPM0020
		COMMON/CA/CGMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,	PDPM0030
		IEPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE	PDPM0040
		COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG	PDPM0050
		COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM	PDPM0060
		COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG	PDPM0070
		COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE	PDPM0080
		COMMON/XYZ/PDPBPI	PDPM0090
		NTYPE=3	PDPM0100
		P=1.0	PDPM0110
		DO 2 I=1,NC	PDPM0120
		TR=I/TC(I)	PDPM0130
A	2	AK(I)=PC(I)/P*DEXP(2.3026*((16.25-73.85*ZC(I)+90.*ZC(I)**2)*(TR-1.	PDPM0140
B	2	AK(I)=PC(I)/P*EXP(2.3026*((16.25-73.85*ZC(I)+90.*ZC(I)**2)*(TR-1.	PDPM0150
		1)/TR-10.*(-8.68*(TR-1.8+6.2*ZC(I))**2)))	PDPM0160
		CALL PDPBPM	PDPM0170
		P=PDPBPI	PDPM0180
		PF=P	PDPM0190
		IFLAG=1	PDPM0200
		CALL DEWBUB	PDPM0210
		RETURN	PDPM0220
		END	PDPM0230

		SUBROUTINE PDPBPM	PDPB0010
A		IMPLICIT REAL*8(A-H,O-Z)	PDPB0020
		COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,	PDPB0030
		IEPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE	PDPB0040
		COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG	PDPB0050
		COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM	PDPB0060
		COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG	PDPB0070
		COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE	PDPB0080
		COMMON/XYZ/PDPBPI	PDPB0090
		PBPI=0.	PDPB0100
		PDPI=0.	PDPB0110
		DO 1 I=1,NC	PDPB0120
		PBPI=PBPI+AK(I)*Z(I)*P	PDPB0130
1		PDPI=PDPI+Z(I)/(AK(I)*P)	PDPB0140
		PDPI=1./PDPI	PDPB0150
		PDPBPI=(PDPI+PBPI)/2.	PDPB0160
		RETURN	PDPB0170
		END	PDPB0180

A		SUBROUTINE PHASE	PHAS0010
		IMPLICIT REAL*8(A-H,O-Z)	PHAS0020


```

COMMON /CA/ COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,PHAS0030
1EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE PHAS0040
COMMON /CB/ A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG PHAS0050
COMMON /CC/ B(2,11),C1(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM PHAS0060
COMMON /CK/ ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG PHAS0070
COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE PHAS0080
DO 10 I=1,NC PHAS0090
FU(I)=1.0 PHAS0100
FUL(I)=1.0 PHAS0110
10 FUG(I)=1.0 PHAS0120
V=0.5 PHAS0130
DO 70 IT=1,ITNM PHAS0140
IF (IPRNT.NE.0) WRITE (6,110) IT PHAS0150
CALL FLASH PHAS0160
IF (IT.LE.NPHASE) GO TO 20 PHAS0170
IF (V.EQ.0.EQ.OR.V.EQ.1.EQ) GO TO 80 PHAS0180
20 MD=-1 PHAS0190
CALL F PHAS0200
MD=0 PHAS0210
CALL F PHAS0220
DO 30 I=1,NC PHAS0230
30 AK(I)=FUL(I)*Y(1)/(X(1)*FUG(I)) PHAS0240
IF (IPRNT.EQ.0) GO TO 50 PHAS0250
WRITE (6,120) PHAS0260
DO 40 I=1,NC PHAS0270
40 WRITE (6,130) (COMP(J,I),J=1,4),Y(I),X(I),AK(I),FUG(I),FUL(I) PHAS0280
50 DO 60 I=1,NC PHAS0290
IF (DABS(FUL(I)/FUG(I)-1.00).GT.FUGERR) GO TO 70 PHAS0300
IF (ABS(FUL(I)/FUG(I)-1.00).GT.FUGERR) GO TO 70 PHAS0310
60 CONTINUE PHAS0320
IF (IT.LT.NPHASE) GO TO 70 PHAS0330
GO TO 80 PHAS0340
70 CONTINUE PHAS0350
WRITE (6,90) ITNM PHAS0360
80 MD=-1 PHAS0370
CALL F PHAS0380
IF (NLG.NE.MD) V=1. PHAS0390
CALL HSCP PHAS0400
HL=HC PHAS0410
SL=SC PHAS0420
CPL=CP PHAS0430
MD=0 PHAS0440
CALL F PHAS0450
IF (NLG.NE.MD) V=0. PHAS0460
CALL HSCP PHAS0470
HG=HC PHAS0480
SG=SC PHAS0490
CPG=CP PHAS0500
HM=V*HG+(1.00-V)*HL PHAS0510
SM=V*SG+(1.00-V)*SL PHAS0520

```

```

CPM=V*CPG+(1.E0-V)*CPL
IF (IPRNT.NE.0) WRITE (6,100) T,P,HM,SM
RETURN

```

```

PHAS0530
PHAS0540
PHAS0550
PHAS0560
PHAS0570
PHAS0580
PHAS0590
PHAS0600
PHAS0610
PHAS0620
PHAS0630

```

```

90 FORMAT(///,1H,25H PHASE FAILED TO CONVERGE: ,15,4H ITS ,/)
100 FORMAT(///,1H,6HT(R) = ,G16.8,10H P(PSIA) = ,G16.8,5H HM = ,G16.8,5H SM = ,G16.8)
110 FORMAT(///,1H,19HBEGIN PHASE CALC. # ,13)
120 FORMAT(24X,1HY ,14X,1HX ,14X,1HK ,14X,3HFUG ,12X,3HFUL )
130 FORMAT (1X,4A4,5G15.8)
END

```

```

SUBROUTINE PROPS(FLAG)
IMPLICIT REAL*8(A-H,O-Z)
SUBROUTINE FOR CALCULATION OF PROPERTIES
INTEGER FLAG
COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,
1EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE
COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG
COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM
COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG
COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE
COMMON/CZ/TB(15),XLMW,CV
COMMON/CY/HIDEAL,SIDEAL
COMMON/H/H,S,ENCOR
IF(NC.EQ.1) GO TO 3
CALL BWRC
3 CONTINUE
CALL DHFP
CALL HSCP
CALL CVHCAP
V=1.0/(DC*XLMW)
H=HC/XLMW
H=H-ENCOR
S=SC/XLMW
CP=CP/XLMW
CV=CV/XLMW
TF=T-459.69
IF(FLAG.NE.0) GO TO 4
WRITE(6,2) TF,P,V,H,S,CP,CV,MD
2 FORMAT(F8.2,3X,F8.3,3X,F9.5,3X,F8.2,3X,F7.4,3X,F7.4,3X,F7.4,3X,I2)
4 RETURN
END

```

```

PROP0010
PROP0020
PROP0030
PROP0040
PROP0050
PROP0060
PROP0070
PROP0080
PROP0090
PROP0100
PROP0110
PROP0120
PROP0130
PROP0140
PROP0150
PROP0160
PROP0170
PROP0180
PROP0190
PROP0200
PROP0210
PROP0220
PROP0230
PROP0240
PROP0250
PROP0260
PROP0270
PROP0280
PROP0290
PROP0300
PROP0310

```

SUBROUTINE PUNOUT

PUN00010

A	IMPLICIT REAL*8(A-H,O-Z)	PUN00020
	COMMON/F1/TGRID(30,30),SGRID(30,30),VGRID(30,30),CPGRID(30,30),CVG	PUN00030
	*RID(30,30),IPUNCH,NPF,NHF	PUN00040
	WRITE(6,950)	PUN00050
950	FORMAT(1H1,12HPUNCH OUTPUT ,///)	PUN00060
	WRITE(6,975)	PUN00070
	IF(IPUNCH.EQ.1)WRITE(7,975)	PUN00080
975	FORMAT(1X,11HTEMPERATURE)	PUN00090
	DO 1001 J=1,NHF	PUN00100
	WRITE(6,5000)(TGRID(I,J),I=1,NPF)	PUN00110
	IF(IPUNCH.EQ.1)WRITE(7,5000)(TGRID(I,J),I=1,NPF)	PUN00120
1001	CONTINUE	PUN00130
	WRITE(6,1050)	PUN00140
	IF(IPUNCH.EQ.1)WRITE(7,1050)	PUN00150
1050	FORMAT(1X,7HENTROPY)	PUN00160
	DO 2001 J=1,NHF	PUN00170
	WRITE(6,5000)(SGRID(I,J),I=1,NPF)	PUN00180
	IF(IPUNCH.EQ.1)WRITE(7,5000)(SGRID(I,J),I=1,NPF)	PUN00190
2001	CONTINUE	PUN00200
	WRITE(6,2050)	PUN00210
	IF(IPUNCH.EQ.1)WRITE(7,2050)	PUN00220
2050	FORMAT(1X,15HSPECIFIC VOLUME)	PUN00230
	DO 3001 J=1,NHF	PUN00240
	WRITE(6,5000)(VGRID(I,J),I=1,NPF)	PUN00250
	IF(IPUNCH.EQ.1)WRITE(7,5000)(VGRID(I,J),I=1,NPF)	PUN00260
3001	CONTINUE	PUN00270
	WRITE(6,3050)	PUN00280
	IF(IPUNCH.EQ.1)WRITE(7,3050)	PUN00290
3050	FORMAT(1X,2HCP)	PUN00300
	DO 4001 J=1,NHF	PUN00310
	WRITE(6,5000)(CPGRID(I,J),I=1,NPF)	PUN00320
	IF(IPUNCH.EQ.1)WRITE(7,5000)(CPGRID(I,J),I=1,NPF)	PUN00330
4001	CONTINUE	PUN00340
5000	FORMAT((5X,2HE ,5(E12.4,1H,)))	PUN00350
	RETURN	PUN00360
	END	PUN00370

C-28

A	SUBROUTINE SERCH (XV)	SERC00010
	IMPLICIT REAL*8(A-H,O-Z)	SERC00020
	COMMON /CA/ COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,	SERC00030
	1EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE	SERC00040
	COMMON /CB/ A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG	SERC00050
	COMMON /CC/ B(2,11),C1(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM	SERC00060
	COMMON /CK/ ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG	SERC00070
	COMMON /CT/ AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE	SERC00080
	STEP1=0.382	SERC00090
	DELTP=0.01	SERC00100

```

      ISGN=1
      SGN=1.
      DO 80 IT=1,ITNM
      IF (IPRNT.NE.0) WRITE (6,280) IT
      CALL KI
      DO 10 I=1,NPHASE
10  CALL FUNC (XV,F1)
30  IF (IT.EQ.1.AND.INLG.EQ.0) SGN=-1.
      IF (SGN.GT.0.EQ) GO TO 50
40  IF (INLG) 70,60,90
50  IF (INLG) 60,70,90
60  IF (ISGN.GT.0) DELTP=2.*DELTP
      XV=XV+SGN*DELTP
      GO TO 80
70  XV=XV-SGN*DELTP
      DELTP=STEP1*DELTP
      XV=XV+SGN*DELTP
      ISGN=-1
80  CONTINUE
      WRITE (6,290) ITNM,T,P,DELTP
      RETURN
90  IFLAG=1
      X1=XV
120  X2=XV*STEP
      IT=2
      IF (IPRNT.NE.0) WRITE (6,310) IT,NTYPE
      CALL FUNC (X2,F2)
      IF (INLG.EQ.1) GO TO 130
      WRITE (6,270) NTYPE,IT,INLG
      WRITE (6,320) T,P
      GO TO 250
130  DO 230 IT=3,ITNM
      IF (DABS(F1-F2).LT.1.D-12) GO TO 250
      IF (ABS(F1-F2).LT.1.E-12) GO TO 250
      XINC=(X1-X2)/(F2-F1)*(F2-FS)
A 150  XINCP=DABS(DEXP(X2+XINC)-DEXP(X2))
B 150  XINCP=ABS( EXP(X2+XINC)- EXP(X2))
A      IF (XINCP.LT.50.DS.OR.XINCP/DEXP(X2).LT.0.500) GO TO 180
B      IF (XINCP.LT.50.EQ.OR.XINCP/ EXP(X2).LT.0.5EG) GO TO 180
      XINC=0.5*XINC
      GO TO 150
180  X3=X2+XINC
      IF (IPRNT.NE.0) WRITE (6,310) IT,NTYPE
      CALL FUNC (X3,F3)
      IF (INLG.EQ.1) GO TO 190
      WRITE (6,270) NTYPE,IT,INLG
      WRITE (6,320) T,P
      GO TO 240
190  DO 200 I=1,NC
A      IF (DABS(FUL(I)/FUG(I)-1.D0).GT.FUGERR) GO TO 220

```

```

SERC0110
SERC0120
SERC0130
SERC0140
SERC0150
SERC0160
SERC0170
SERC0180
SERC0190
SERC0200
SERC0210
SERC0220
SERC0230
SERC0240
SERC0250
SERC0260
SERC0270
SERC0280
SERC0290
SERC0300
SERC0310
SERC0320
SERC0330
SERC0340
SERC0350
SERC0360
SERC0370
SERC0380
SERC0390
SERC0400
SERC0410
SERC0420
SERC0430
SERC0440
SERC0450
SERC0460
SERC0470
SERC0480
SERC0490
SERC0500
SERC0510
SERC0520
SERC0530
SERC0540
SERC0550
SERC0560
SERC0570
SERC0580
SERC0590
SERC0600

```

B	IF (ABS(FUL(I)/FUG(I)-1.E0).GT.FUGERR) GO TO 220	SERC0610
200	CONTINUE	SERC0620
A	IF (DABS(F3).LE.EPSS) GO TO 240	SERC0630
B	IF (ABS(F3).LE.EPSS) GO TO 240	SERC0640
220	X1=X2	SERC0650
	X2=X3	SERC0660
	F1=F2	SERC0670
	F2=F3	SERC0680
230	CONTINUE	SERC0690
	WRITE (6,300) ITNM	SERC0700
240	XV=X3	SERC0710
	RETURN	SERC0720
250	XV=X2	SERC0730
260	RETURN	SERC0740
C		SERC0750
270	FORMAT(//,1H ,29HDEWBUB SEARCH FAILED(NTYPE = ,I3,7H;ITS = ,I3,8H;INLG = ,I3,53H :(-1=LIQUID DENSITY FAILED; 0=VAPOR DENSITY FAILED)))	SERC0760
280	FORMAT(//,1H ,44HBEGIN TRIAL AND ERROR SEARCH(KK.NE.1)CALC. # ,I3)	SERC0770
290	FORMAT(////,1H ,42H****DEWBUB TRIAL AND ERROR SEARCH FAILED(,I3,14H) ITS: T(R) = ,G13.6,5H P = ,G13.6,9H DELTP = ,G13.6)	SERC0780
300	FORMAT(////,1H ,26HSEARCH FAILED TO CONVERGE: ,I5,4H ITS ,/)	SERC0790
310	FORMAT(////,1H ,28HBEGIN SEARCH CALCULATION NO. ,I4,9H NTYPE = ,I2)	SERC0800
320	FORMAT(10X,14HTEMPERATURE = ,F15.6,10X,11HPRESSURE = ,F15.6)	SERC0810
	END	SERC0820
		SERC0830
		SERC0840
		SERC0850
		SERC0860

C-30

	SUBROUTINE TDPBPM(TDPI,TBPI,PHIII)	TDPB0010
A	IMPLICIT REAL*8(A-H,O-Z)	TDPB0020
	COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,	TDPB0030
1	EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE	TDPB0040
	COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG	TDPB0050
	COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE	TDPB0060
	COMMON/CZ/TB(15),XLMW,CV	TDPB0070
	ZMAX=0.0	TDPB0080
	DO 10 I=1,NC	TDPB0090
	IF(Z(I) .LT. ZMAX) GO TO 10	TDPB0100
	ZMAX=Z(I)	TDPB0110
	IZMAX=I	TDPB0120
10	CONTINUE	TDPB0130
	I=IZMAX	TDPB0140
	PB=14.696	TDPB0150
A	ZB=(DLOG(PC(I))-DLOG(PB))/(1./TB(I)-1./TC(I))	TDPB0160
B	ZB=(ALOG(PC(I)) - ALOG(PB))/(1./TB(I) - 1./TC(I))	TDPB0170
A	ZA=ZB/TB(I)+DLOG(PB/P)	TDPB0180
B	ZA=ZB/TB(I) +ALOG(PB/P)	TDPB0190
	T=ZB/ZA	TDPB0200

```

30 PBPI=0.
DO 2 I=1,NC
TR=T/TC(I)
A AK(I)=PC(I)/P*DEXP(2.3026*((16.25-73.85*ZC(I)+90.*ZC(I)**2)*(TR-1.
B AK(I)=PC(I)/P* EXP(2.3026*((16.25-73.85*ZC(I)+90.*ZC(I)**2)*(TR-1.
1)/TR-10.**(-8.68*(TR-1.8+6.2*ZC(I))**2)))
2 PBPI=PBPI + AK(I)*Z(I)*P
P=PBPI
CALL ITER(5,T,P,PHIII,FLAG)
IF(FLAG.NE.0.0) GO TO 30
TBPI=T
40 PDPI=0.0
DO 5 I=1,NC
TR=T/TC(I)
A AK(I)=PC(I)/P*DEXP(2.3026*((16.25-73.85*ZC(I)+90.*ZC(I)**2)*(TR-1.
B AK(I)=PC(I)/P* EXP(2.3026*((16.25-73.85*ZC(I)+90.*ZC(I)**2)*(TR-1.
1)/TR-10.**(-8.68*(TR-1.8+6.2*ZC(I))**2)))
5 PDPI=PDPI + Z(I)/(AK(I)*P)
P=1.0/PDPI
CALL ITER(6,T,P,PHIII,FLAG)
IF(FLAG.NE.0.0) GO TO 40
TDPI=T
RETURN
END

```

TDPB0210
TDPB0220
TDPB0230
TDPB0240
TDPB0250
TDPB0260
TDPB0270
TDPB0280
TDPB0290
TDPB0300
TDPB0310
TDPB0320
TDPB0330
TDPB0340
TDPB0350
TDPB0360
TDPB0370
TDPB0380
TDPB0390
TDPB0400
TDPB0410
TDPB0420
TDPB0430
TDPB0440

C-37

```

SUBROUTINE VAPRES
IMPLICIT REAL*8(A-H,O-Z)
SUBROUTINE FOR CALCULATION OF VAPOR PRESSURE
COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,
1 EPSV,EPS3,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE
COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG
COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM
COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG
COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE
COMMON/CZ/TB(15),XLMW,CV
COMMON/CY/HIDEAL,SIDEAL
PB=14.696
A ZB=(DLOG(PC(1))-DLOG(PB))/(1./TB(1)-1./TC(1))
B ZB=(ALOG(PC(1))-ALOG(PB))/(1./TB(1)-1./TC(1))
A ZA=DLOG(PB)+ZB/TB(1)
B ZA=ALOG(PB)+ZB/TB(1)
A P=DEXP(ZA-ZB/T)
B P= EXP(ZA-ZB/T)
J=1
12 MD=0
CALL DHFP
CALL HSCP

```

VAPR0010
VAPR0020
VAPR0030
VAPR0040
VAPR0050
VAPR0060
VAPR0070
VAPR0080
VAPR0090
VAPR0100
VAPR0110
VAPR0120
VAPR0130
VAPR0140
VAPR0150
VAPR0160
VAPR0170
VAPR0180
VAPR0190
VAPR0200
VAPR0210
VAPR0220

A	FUG(1)=DEXP((HC-HIDEAL-T*(SC-SIDEAL))/(R*T*CONV))	VAPR0230
B	FUG(1)= EXP((HC-HIDEAL-T*(SC-SIDEAL))/(R*T*CONV))	VAPR0240
	MD=-1	VAPR0250
	CALL DHFP	VAPR0260
	CALL HSCP	VAPR0270
A	FUL(1)=DEXP((HC-HIDEAL-T*(SC-SIDEAL))/(R*T*CONV))	VAPR0280
B	FUL(1)= EXP((HC-HIDEAL-T*(SC-SIDEAL))/(R*T*CONV))	VAPR0290
A	IF(DABS((FUL(1)-FUG(1))/FUG(1))-0.0001)15,15,11	VAPR0300
B	IF(ABS((FUL(1)-FUG(1))/FUG(1))-0.0001)15,15,11	VAPR0310
11	P=P*FUL(1)/FUG(1)	VAPR0320
	J=J+1	VAPR0330
	IF(J-20)12,10,10	VAPR0340
10	CONTINUE	VAPR0350
	WRITE(6,14) T,P,FUG(1),FUL(1)	VAPR0360
14	FORMAT(7,1H ,41HVAPOR PRESS. NOT CONVERGED AT T P FUG FUL ,2X,4F9	VAPR0370
	*.4,7)	VAPR0380
15	CONTINUE	VAPR0390
	RETURN	VAPR0400
	END	VAPR0410

	SUBROUTINE VATEMP	VATE0010
A	IMPLICIT REAL*8(A-H,O-Z)	VATE0020
C***	SUBROUTINE FOR CALCULATION OF VAPOR TEMPERATURE	VATE0030
	COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,OL,FS,	VATE0040
1	EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE	VATE0050
	COMMON/CB/A(15,11),CK1J(15,15),FU(15),NPARM,IFLAG	VATE0060
	COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM	VATE0070
	COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX ,INLG,NLG	VATE0080
	COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE	VATE0090
	COMMON/CZ/TB(15),XLMW,CV	VATE0100
	COMMON/CY/HIDEAL,SIDEAL	VATE0110
	DIMENSION TEMP(30),FUGAC(30),FULAC(30)	VATE0120
	PB=14.696	VATE0130
A	ZB=(DLOG(PC(1))-DLOG(PB))/(1./TB(1)-1./TC(1))	VATE0140
B	ZB=(ALOG(PC(1))-ALOG(PB))/(1./TB(1)-1./TC(1))	VATE0150
A	ZA=ZB/TB(1)+DLOG(PB/P)	VATE0160
B	ZA=ZB/TB(1)+ALOG(PB/P)	VATE0170
	T=ZB/ZA	VATE0180
	TEMP(1)=T	VATE0190
	DO 200 I=1,20	VATE0200
	T=TEMP(I)	VATE0210
12	MD=0	VATE0220
	CALL DHFP	VATE0230
	CALL HSCP	VATE0240
A	FUG(1)=DEXP((HC-HIDEAL-TEMP(I)*(SC-SIDEAL))/(R*TEMP(I)*CONV))	VATE0250
B	FUG(1)= EXP((HC-HIDEAL-TEMP(I)*(SC-SIDEAL))/(R*TEMP(I)*CONV))	VATE0260
	FUGAC(1)=FUG(1)	VATE0270

```

MD=-1
CALL DHFP
CALL HSCP
FUL(1)=DEXP((HC-HIDEAL-TEMP(I)*(SC-SIDEAL))/(R*TEMP(I)*CONV))
FUL(1)=EXP((HC-HIDEAL-TEMP(I)*(SC-SIDEAL))/(R*TEMP(I)*CONV))
FULAC(I)=FUL(1)
IF(DABS((FULAC(I)-FUGAC(I))/FUGAC(I))-0.0001)15,15,11
IF(ABS((FULAC(I)-FUGAC(I))/FUGAC(I))-0.0001)15,15,11
11 IF(I.EQ.1) GO TO 210
SLOPE=(FUGAC(I)-FULAC(I)-FUGAC(I-1)+FULAC(I-1))/(TEMP(I)-TEMP(I-1))
*)
TEMP(I+1)=(SLOPE*TEMP(I)-FUGAC(I)+FULAC(I))/SLOPE
GO TO 200
210 TEMP(2)=TEMP(1)+1.0
200 CONTINUE
WRITE(6,14) T,P,FUG(1),FUL(1)
14 FORMAT(/,1H,41HVAPOR PRESS. NOT CONVERGED AT T P FUG FUL ,2X,4F9
*.4,/)
15 CONTINUE
RETURN
END

```

VATE0280
VATE0290
VATE0300
VATE0310
VATE0320
VATE0330
VATE0340
VATE0350
VATE0360
VATE0370
VATE0380
VATE0390
VATE0400
VATE0410
VATE0420
VATE0430
VATE0440
VATE0450
VATE0460
VATE0470
VATE0480

```

SUBROUTINE VLREGM(FLAG)
IMPLICIT REAL*8(A-H,O-Z)
INTEGER FLAG
COMMON/CA/COMP(4,15),FUG(15),FUL(15),X(15),Y(15),Z(15),DG,DL,FS,
1EPSV,EPSS,FUGERR,HG,HL,HM,PF,SG,SL,SM,STEP,TF,V,NPHASE,NTYPE
COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,IFLAG
COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM
COMMON/CK/ACF(15),CD(15),ZC(15),PC(15),TC(15),EPSD,DMAX,INLG,NLG
COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE
COMMON/CZ/TB(15),XLMW,CV
COMMON/CX/VOLM,HMM,SMM,CPMM,CVMM
COMMON/CY/HIDEAL,SIDEAL
COMMON/F2/IGRID,IDMPRT
COMMON/H/H,S,ENCOR
CALL KI
CALL PHASE
C** PROPERTIES OF TWO PHASE MIXTURE
MD=2
CVMM=0.0
HMM=HM/XLMW
HMM=HMM-ENCOR
SMM=SM/XLMW
CPMM=CPM/XLMW
VOLM=(V/DG+(1.-V)/DL)/XLMW
IF(FLAG.NE.C) GO TO 4

```

VLRG0010
VLRG0020
VLRG0030
VLRG0040
VLRG0050
VLRG0060
VLRG0070
VLRG0080
VLRG0090
VLRG0100
VLRG0110
VLRG0120
VLRG0130
VLRG0140
VLRG0150
VLRG0160
VLRG0170
VLRG0180
VLRG0190
VLRG0200
VLRG0210
VLRG0220
VLRG0230
VLRG0240
VLRG0250

CALL DOMJOUT
4 RETURN
END

VLRG0260
VLRG0270
VLRG0280

	BLOCK DATA		BLDA0010
A	IMPLICIT REAL*8(A-H,O-Z)		BLDA0020
A	REAL*8 ISWT,MINX,ICNT,MI,MAXX,ITF,NI		BLDA0030
B	REAL ISWT,MINX,ICNT,MI,MAXX,ITF,NI		BLDA0040
	COMMON/CB/A(15,11),CKIJ(15,15),FU(15),NPARM,I FLAG		BLDA0050
	COMMON/CC/B(2,11),CI(15,7),CMW(15),CONV,HC,SC,CP,CPL,CPG,CPM		BLDA0060
	COMMON/CT/AK(15),AL(11),XL(15),DC,P,R,T,IPRNT,ITNM,MD,NC,IPHASE		BLDA0070
	COMMON/ITERS/ISWT(30)		BLDA0080
	COMMON/ITERS1/EPS(30),MINX(30),MAXX(30),PCNT1(30),PCNTN(30),		BLDA0090
	1ICNT(30),ITF(30),XNAM(30),YNAM(30),MI,NI		BLDA0100
	DATA CONV,R,NPARM/0.185057E0,.107335E2,11/		BLDA0110
	DATA B(1,1),B(1,2),B(1,3),B(1,4),B(1,5),B(1,6),B(1,7),B(1,8),B(1,9)		BLDA0120
	*,B(1,10),B(1,11)/0.44369E0,1.2844E0,0.35631E0,0.54498E0,0.52863E0		BLDA0130
	*,0.484C1E0,0.070523E0,0.50409E0,0.030745E0,0.073283E0,0.00645E0/		BLDA0140
	DATA B(2,1),B(2,2),B(2,3),B(2,4),B(2,5),B(2,6),B(2,7),B(2,8),B(2,9)		BLDA0150
	*,B(2,10),B(2,11)/0.11545E0,-0.92073E0,1.7087E0,-0.2709E0,0.34926		BLDA0160
	*E0,0.75413E0,-0.044448E0,1.3224E0,0.17943E0,0.46349E0,-0.022143E0/		BLDA0170
	DATA ISWT/30*0.E0/		BLDA0180
	DATA EPS/4*5.E-5,2*5.E-2,24*5.E-5/		BLDA0190
	DATA MINX/30*0.0/		BLDA0200
	DATA MAXX/30*2.E3/		BLDA0210
	DATA PCNT1/5.E-2,3*5.E-3,2*1.E-3,24*5.E-5/		BLDA0220
	DATA PCNTN/30*1.E0/		BLDA0230
	DATA ICNT/30*0.E0/		BLDA0240
	DATA ITF/30*1.E0/		BLDA0250
	DATA XNAM/6HTGUESS,6H TDPM ,6H TBPM ,6H TVLR ,6HTBPGES,6HTDPGES		BLDA0260
	*,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE		BLDA0270
	*,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE		BLDA0280
	*,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE		BLDA0290
	*,6H NONE ,6H NONE ,6H NONE /		BLDA0300
	DATA YNAM/6HHGUESS,6H PDPM ,6H PBPM ,6H HVLR ,6H PRES ,6H PRES		BLDA0310
	*,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE		BLDA0320
	*,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE		BLDA0330
	*,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE ,6H NONE		BLDA0340
	*,6H NONE ,6H NONE ,6H NONE /		BLDA0350
	DATA MI/9.E0/		BLDA0360
	DATA NI/19.E0/		BLDA0370
	END		BLDA0380

*****INPUT CARD ECHO*****

00000000111111112222222233333333444444445555555566666666777777778
1234567890123456789012345678901234567890123456789012345678901234567890

1	0	1	3	20	0	0	0	0	0	0	0	0
2	2.0	0.1E-6	0.2E-6	0.3E-6	0.1E-4	1.0001	0.0	0.0	0.0	0.0	0.0	0.0
3	450.	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	ISOBUTANE	1	58.12	274.96	0.183	0.2373	529.0999	10.90	10.90	10.90	10.90	10.90
5	13.2865990	0.36637	0.349631E-0030	0.536100E-008	0.298111E-100	0.538662E-0140	0.60935	0.60935	0.60935	0.60935	0.60935	0.60935
6	1.87890	37264.0	0.101413E11	7.11486	8.58663	47990.7	47990.7	47990.7	47990.7	47990.7	47990.7	47990.7
7	4.23987	0.406763E11	0.853176E12	0.216863E8	0.840860E14	0.840860E14	0.840860E14	0.840860E14	0.840860E14	0.840860E14	0.840860E14	0.840860E14
8	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
9	5	10	150.	200.	300.	400.	500.	600.	700.	700.	700.	700.
10	100.	150.	200.	300.	400.	500.	600.	700.	700.	700.	700.	700.
11	100.	150.	200.	300.	400.	500.	600.	700.	700.	700.	700.	700.
12	800.	900.	900.	900.	900.	900.	900.	900.	900.	900.	900.	900.
13	0	2	3	20	0	0	1	0	0	0	0	0
14	2.0	0.1E-6	0.2E-6	0.3E-6	0.1E-4	1.0001	0.0	0.0	0.0	0.0	0.0	0.0
15	450.	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
16	ISOBUTANE	1	58.12	274.96	0.183	0.2373	529.0999	10.90	10.90	10.90	10.90	10.90
17	13.2865990	0.36637	0.349631E-0030	0.536100E-008	0.298111E-100	0.538662E-0140	0.60935	0.60935	0.60935	0.60935	0.60935	0.60935
18	PROPANE	2	44.094	206.13	0.157	0.3121	617.4	-43.70	-43.70	-43.70	-43.70	-43.70
19	-1.22301	0.179733	0.066458E-03	0.250998E-06	1.247461E-10	1.893509E-14	0.178189	0.178189	0.178189	0.178189	0.178189	0.178189
20	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003
21	0.75	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25
22	10	5	10	40.	70.	100.	130.	160.	190.	190.	190.	190.
23	-20.	10.	40.	70.	100.	130.	160.	190.	190.	190.	190.	190.
24	220.	250.	250.	250.	250.	250.	250.	250.	250.	250.	250.	250.
25	200.	300.	400.	700.	800.	800.	800.	800.	800.	800.	800.	800.

*****EDIT THIS CASE*****

IGRID,NC,NPHASE,ITNM,IPRNT,IDMPRT,IPUNCH,IPNPR
 0 1 3 20 0 0 0 C

DMAX,EPSP,EPSPV,EUGERR,EPSS,STEP,ENCDR
 .200000E+01 .100000E-06 .200000E-06 .300000E-06 .100000E-04 .100010E+01 0.

TPHIN,(ISWT(I),I=1,6)
 450.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000

(COMP(J,I),J=1,4),IDCOM(1),CMW(1),TC(1),ACF(I),CD(I),PC(1),TB(1)

ISOBUTANE 1 55.1200 734.6500 .1830 .2373 529.0999 470.5900
 CI .132866E+02 .366370E-01 .349631E-03 .536100E-08 -.298111E-10 .538662E-14 .609350E+00

((IA(I,J),J=1,NPARM),I=1,NC)
 .187890E+01 .372640E+05 .101413E+11 .711486E+01 .858663E+01 .479907E+05 .423987E+01
 .406763E+11 .853176E+12 .216863E+08 .840860E+14

CKIJ(I,J)

0.

COMPONENT MOLE FRACTIONS

ISOBUTANE 1.00000

NUM1 AND NUM2 5 10

TEMPERATURE OR ENTHALPY GRID LINES

100.00 150.00 200.00 300.00 400.00

PRESSURE GRID LINES

100.00 150.00 200.00 300.00 400.00 500.00 600.00 700.00 800.00 900.00

PRESSURE - ENTHALPY GRID

TEMPERATURES IN F
 PRESSURES IN PSIA
 SPECIFIC VOLUMES IN CUBIC FEET PER POUND
 ENTHALPIES IN BTU/LB
 ENTROPIES IN BTU/LB-R
 SPECIFIC HEATS IN BTU/LB-R
 M=-1 IMPLIES LIQUID REGION
 M=0 IMPLIES VAPOR REGION
 M=2 IMPLIES 2-PHASE REGION

SATURATED PROPERTIES

C-37	TF	P	V	H	S	CP	CV	M
	121.72	100.000	.91030	141.06	1.1674	.4759	0.0000	0
	121.72	100.000	.03076	10.97	1.9436	.6247	0.0000	-1
	152.84	150.000	.60161	150.74	1.1722	.5173	0.0000	0
	152.84	150.000	.03248	31.11	.9769	.6746	0.0000	-1
	177.02	200.000	.44165	157.83	1.1759	.5587	0.0000	0
	177.02	200.000	.03423	47.90	1.0033	.7310	0.0000	-1
	214.47	300.000	.27427	167.30	1.1806	.6638	0.0000	0
	214.47	300.000	.03843	76.97	1.0466	.8964	0.0000	-1
	243.84	400.000	.18336	171.97	1.1813	.8851	0.0000	0
	243.84	400.000	.04506	104.59	1.0855	1.2723	0.0000	-1
	268.74	500.000	.11756	169.76	1.1744	2.3810	0.0000	0
	268.74	500.000	.06002	136.58	1.1288	3.3021	0.0000	-1

GRID PROPERTIES

TF	P	V	H	S	CP	CV	M
121.72	100.000	.633	100.00	1.0968	*****	*****	2
140.44	100.000	.95847	150.00	1.1825	.4804	0.0000	0
240.65	100.000	1.19235	200.00	1.2595	.5203	0.0000	0
418.15	100.000	1.56259	300.00	1.3864	.6070	0.0000	0
573.38	100.000	1.86882	400.00	1.4911	.6805	0.0000	0

C-38

TF	P	V	H	S	CP	CV	M
152.84	150.000	.360	100.00	1.0894	*****	*****	2
152.84	150.000	.598	150.00	1.1710	*****	*****	2
246.94	150.000	.76790	200.00	1.2470	.5355	0.0000	0
421.23	150.000	1.02603	300.00	1.3731	.6135	0.0000	0
575.29	150.000	1.23550	400.00	1.4776	.6841	0.0000	0

TF	P	V	H	S	CP	CV	M
177.02	200.000	.227	100.00	1.0851	*****	*****	2
177.02	200.000	.413	150.00	1.1636	*****	*****	2
253.41	200.000	.5527	200.00	1.2385	.5551	0.0000	0
424.33	200.000	.75784	300.00	1.3639	.6202	0.0000	0
577.21	200.000	.91899	400.00	1.4681	.6878	0.0000	0

C-39

TF	P	V	H	S	CP	CV	M
214.47	300.000	.099	100.00	1.0807	*****	*****	2
214.47	300.000	.229	150.00	1.1549	*****	*****	2
267.00	300.000	.34183	199.99	1.2273	.6027	0.0000	0
430.63	300.000	.48989	300.00	1.3513	.6349	0.0000	0
581.06	300.000	.60285	400.00	1.4549	.6954	0.0000	0

TF	P	V	H	S	CP	CV	M
240.04	400.000	.04342	100.00	1.0790	1.1476	0.0000	-1
243.84	400.000	.138	150.00	1.1501	*****	*****	2
281.68	400.000	.23442	200.00	1.2202	.6745	0.0000	0
437.02	400.000	.35628	300.00	1.3427	.6511	0.0000	0
584.91	400.000	.44520	400.00	1.4458	.7034	0.0000	0

C-40

TF	P	V	H	S	CP	CV	M
243.19	500.000	.04203	100.00	1.0779	1.0093	0.0000	-1
268.74	500.000	.083	150.00	1.1473	*****	*****	2
297.49	500.000	.17014	200.00	1.2152	.7909	0.0000	0
443.48	500.000	.27654	300.00	1.3362	.6689	0.0000	0
588.74	500.000	.35100	400.00	1.4388	.7118	0.0000	0

TF	P	V	H	S	CP	CV	M
245.67	600.000	.04094	100.00	1.0768	.9308	0.0000	0
285.19	600.000	.06380	150.00	1.1455	2.1823	0.0000	0
313.83	600.000	.12887	200.00	1.2116	.9617	0.0000	0
449.96	600.000	.22389	300.00	1.3311	.6883	0.0000	0
592.55	600.000	.28856	400.00	1.4332	.7204	0.0000	0

TF	P	V	H	S	CP	CV	M
247.69	700.000	.04004	100.00	1.0757	.8796	0.0000	0
294.13	700.000	.05767	150.00	1.1440	1.3880	0.0000	0
328.96	700.000	.10262	200.00	1.2089	1.1028	0.0000	0
456.38	700.000	.18684	300.00	1.3270	.7089	0.0000	0
596.29	700.000	.24432	399.99	1.4285	.7292	0.0000	0

TF P V H S CP CV M

C-42

249.36	800.000	.03929	100.00	1.0747	.8432	0.0000	0
300.32	800.000	.05380	150.00	1.1427	1.1549	0.0000	0
341.30	800.000	.08640	200.00	1.2067	1.1168	0.0000	0
462.67	800.000	.15966	300.00	1.3235	.7298	0.0000	0
600.00	800.000	.21148	400.00	1.4246	.7380	0.0000	0

TF	P	V	H	S	CP	CV	M
250.76	900.000	.03853	100.00	1.0737	.8159	0.0000	0
304.99	900.000	.05104	150.00	1.1414	1.0409	0.0000	0
350.77	900.000	.07614	200.00	1.2049	1.0651	0.0000	0
468.71	900.000	.13916	300.00	1.3205	.7499	0.0000	0
603.60	900.000	.18627	400.00	1.4211	.7468	0.0000	0

*****EDIT THIS CASE*****

IGRID,NC,NPHASE,IINM,IPRNT,IDMPRI,IPUNCH,IPNPR

DMAX,EPSD,EPSV,FUGER,EPSS,STEP,ENCOR

.200000E+01 .100000E-06 .200000E-06 .300000E-06 .100000E-04 .100010E+01 0.

TPHIN,(ISWT(I),I=1,6)

450.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000

(COMP(J,I),J=1,4),IDCOM(I),CMW(I),IC(I),ACF(I),CD(I),PC(I),IB(I)

ISOBUTANE 1 58.1200 734.6500 .1830 .2373 529.0999 470.5900

CI .132866E+02 .366370E-01 .349631E-03 .536100E-08 -.298111E-10 .538662E-14 .609350E+00

PROPANE 2 44.0940 665.8200 .1570 .3121 617.4000 415.9900

CI -.122301E+01 .179733E+00 .664580E-04 .250998E-06 -.124746E-09 .189351E-13 .178189E+00

((A(I,J),J=1,NPARM),I=1,NC)

.195878E+01	.370810E+05	.119981E+11	.879752E+01	.105227E+02	.871019E+05	.466890E+01
.563869E+11	.837704E+12	.162646E+05	.426647E+14	.147970E+01	.261006E+05	.634021E+10
.515827E+01	.598999E+01	.441979E+05	.209025E+01	.231486E+11	.398203E+12	.713464E+07
.204110E+14						

CKIJ(I,J)

0. .30000E-02

COMPONENT MOLE FRACTIONS

ISOBUTANE .75000

PROPANE .25000

NUM1 AND NUM2 10 5

TEMPERATURE OR ENTHALPY GRID LINES

-20.00 10.00 40.00 70.00 100.00 130.00 160.00 190.00 220.00 250.00

PRESSURE GRID LINES

200.00 300.00 400.00 700.00 800.00

PRESSURE - ENTHALPY GRID

TEMPERATURES IN F
 PRESSURES IN PSIA
 SPECIFIC VOLUMES IN CUBIC FEET PER POUND
 ENTHALPIES IN BTU/LB
 ENTROPIES IN BTU/LB-R
 SPECIFIC HEATS IN BTU/LB-R
 M=-1 IMPLIES LIQUID REGION
 M=0 IMPLIES VAPOR REGION
 M=2 IMPLIES 2-PHASE REGION

SATURATED PROPERTIES

C-44	TF	P	V	H	S	CP	CV	M
	165.95	200.000	.46884	151.84	1.2252	.5432	0.0000	0
	156.66	200.000	.03358	33.96	1.0337	.6756	0.0000	-1
	201.47	300.000	.29482	159.99	1.2272	.6347	0.0000	0
	193.67	300.000	.03691	60.58	1.0746	.8157	0.0000	-1
	228.36	400.000	.20188	163.74	1.2261	.7837	0.0000	0
	222.26	400.000	.04228	86.12	1.1118	1.2722	0.0000	-1

GRID PROPERTIES

TF	P	V	H	S	CP	CV	M
66.84	200.000	.02915	-20.00	.9394	.5497	0.0000	-1
119.01	200.000	.03131	10.00	.9936	.6050	0.0000	-1
157.26	200.000	.05465	40.00	1.0485	.6716	0.0000	2
160.09	200.000	.16124	70.00	1.0971	.6438	0.0000	2
162.53	200.000	.25830	100.00	1.1454	.6152	0.0000	2
164.58	200.000	.37597	130.00	1.1935	.5858	0.0000	2
178.37	200.000	.48512	160.00	1.2406	.5503	0.0000	0
233.28	200.000	.57103	190.00	1.2857	.5471	0.0000	0
287.50	200.000	.64534	220.00	1.3274	.5612	0.0000	0
340.07	200.000	.71195	250.00	1.3662	.5807	0.0000	0

C-45

TF	P	V	H	S	CP	CV	M
67.00	300.000	.02942	-20.00	.9421	.5545	0.0000	-1
118.88	300.000	.03149	10.00	.9964	.6058	0.0000	-1
165.63	300.000	.03423	40.00	1.0462	.6872	0.0000	-1
194.57	300.000	.06001	70.00	1.0941	.7823	0.0000	2
197.20	300.000	.13383	100.00	1.1398	.7473	0.0000	2
199.47	300.000	.20799	130.00	1.1854	.7108	0.0000	2
200.12	300.000	.28690	160.00	1.2297	.6615	0.0000	0
248.52	300.000	.35332	190.00	1.2736	.5979	0.0000	0
299.12	300.000	.40843	220.00	1.3145	.5923	0.0000	0
349.40	300.000	.45670	250.00	1.3528	.6024	0.0000	0

	TF	P	V	H	S	CP	CV	M
	66.55	400.000	.02930	-20.00	.9407	.5519	0.0000	-1
	118.74	400.000	.03133	10.00	.9950	.6013	0.0000	-1
	165.99	400.000	.03398	40.00	1.0448	.6767	0.0000	-1
	206.38	400.000	.03806	70.00	1.0912	.8448	0.0000	-1
	223.63	400.000	.06995	100.00	1.1371	1.0220	0.0000	2
	226.04	400.000	.12623	130.00	1.1809	.9650	0.0000	2
	228.12	400.000	.18276	160.00	1.2246	.9030	0.0000	2
	263.08	400.000	.24074	190.00	1.2672	.6828	0.0000	0
	308.93	400.000	.28696	220.00	1.3074	.6379	0.0000	0
	356.25	400.000	.32636	249.99	1.3453	.6329	0.0000	0

C-46

TF	P	V	H	S	CP	CV	M
65.56	700.000	.02924	-20.00	.9390	.5499	0.0000	0
118.12	700.000	.03112	10.00	.9934	.5940	0.0000	0
166.37	700.000	.03348	40.00	1.0433	.6543	0.0000	0
209.40	700.000	.03676	70.00	1.0896	.7524	0.0000	0
245.04	700.000	.04219	100.00	1.1332	.9799	0.0000	0
269.51	700.000	.05378	130.00	1.1750	1.5635	0.0000	0
287.70	700.000	.07659	160.01	1.2157	1.5897	0.0000	0
310.74	700.000	.10831	190.00	1.2552	1.0565	0.0000	0
343.94	700.000	.13914	220.00	1.2934	.8090	0.0000	0
383.27	700.000	.16584	250.00	1.3298	.7322	0.0000	0

C-47

TF	P	V	H	S	CP	CV	M
65.13	800.000	.02917	-20.00	.9380	.5484	0.0000	0
117.88	800.000	.03102	10.00	.9924	.5912	0.0000	0
166.45	800.000	.03330	40.00	1.0423	.6484	0.0000	0
210.09	800.000	.03641	70.00	1.0886	.7358	0.0000	0
247.24	800.000	.04130	100.00	1.1321	.9085	0.0000	0
275.46	800.000	.05063	130.00	1.1737	1.2552	0.0000	0
298.10	800.000	.06765	160.00	1.2139	1.3169	0.0000	0
322.97	800.000	.09150	190.00	1.2529	1.0769	0.0000	0
354.48	800.000	.11718	219.99	1.2905	.8596	0.0000	0
391.77	800.000	.14072	250.00	1.3265	.7663	0.0000	0

PUNCH OUTPUT

TEMPERATURE

E	.6684E+02,	.6700E+02,	.6655E+02,	.6556E+02,	.6513E+02,
E	.1190E+03,	.1189E+03,	.1187E+03,	.1181E+03,	.1179E+03,
E	.1573E+03,	.1656E+03,	.1660E+03,	.1664E+03,	.1664E+03,
E	.1601E+03,	.1945E+03,	.2054E+03,	.2094E+03,	.2101E+03,
E	.1625E+03,	.1972E+03,	.2236E+03,	.2450E+03,	.2472E+03,
E	.1646E+03,	.1995E+03,	.2260E+03,	.2695E+03,	.2755E+03,
E	.1784E+03,	.2001E+03,	.2281E+03,	.2877E+03,	.2981E+03,
E	.2333E+03,	.2485E+03,	.2631E+03,	.3107E+03,	.3230E+03,
E	.2875E+03,	.2991E+03,	.3089E+03,	.3439E+03,	.3545E+03,
E	.3401E+03,	.3494E+03,	.3563E+03,	.3833E+03,	.3918E+03,

ENTROPY

E	.9394E+00,	.9421E+00,	.9407E+00,	.9390E+00,	.9380E+00,
E	.9936E+00,	.9964E+00,	.9950E+00,	.9934E+00,	.9924E+00,
E	.1049E+01,	.1046E+01,	.1045E+01,	.1043E+01,	.1042E+01,
E	.1097E+01,	.1094E+01,	.1091E+01,	.1090E+01,	.1089E+01,
E	.1145E+01,	.1140E+01,	.1137E+01,	.1133E+01,	.1132E+01,
E	.1193E+01,	.1185E+01,	.1181E+01,	.1175E+01,	.1174E+01,
E	.1241E+01,	.1230E+01,	.1225E+01,	.1216E+01,	.1214E+01,
E	.1286E+01,	.1274E+01,	.1267E+01,	.1255E+01,	.1253E+01,
E	.1327E+01,	.1315E+01,	.1307E+01,	.1293E+01,	.1290E+01,
E	.1366E+01,	.1353E+01,	.1345E+01,	.1330E+01,	.1327E+01,

SPECIFIC VOLUME

E	.2915E-01,	.2942E-01,	.2930E-01,	.2924E-01,	.2917E-01,
E	.3131E-01,	.3149E-01,	.3133E-01,	.3112E-01,	.3102E-01,
E	.3465E-01,	.3423E-01,	.3398E-01,	.3348E-01,	.3330E-01,
E	.1612E+00,	.6001E-01,	.3806E-01,	.3676E-01,	.3641E-01,
E	.2684E+00,	.1338E+00,	.6995E-01,	.4219E-01,	.4130E-01,
E	.3760E+00,	.2080E+00,	.1262E+00,	.5378E-01,	.5063E-01,
E	.4851E+00,	.2869E+00,	.1828E+00,	.7659E-01,	.6765E-01,
E	.5710E+00,	.3533E+00,	.2407E+00,	.1083E+00,	.9150E-01,
E	.6453E+00,	.4084E+00,	.2870E+00,	.1391E+00,	.1172E+00,
E	.7119E+00,	.4567E+00,	.3264E+00,	.1658E+00,	.1407E+00,

CP

E	.5497E+00,	.5545E+00,	.5519E+00,	.5499E+00,	.5484E+00,
E	.6050E+00,	.6058E+00,	.6013E+00,	.5940E+00,	.5912E+00,
E	.6716E+00,	.6872E+00,	.6767E+00,	.6543E+00,	.6484E+00,
E	.6438E+00,	.7823E+00,	.8448E+00,	.7524E+00,	.7358E+00,
E	.6152E+00,	.7473E+00,	.1022E+01,	.9799E+00,	.9085E+00,
E	.5858E+00,	.7108E+00,	.9650E+00,	.1563E+01,	.1255E+01,
E	.5503E+00,	.6615E+00,	.9030E+00,	.1590E+01,	.1317E+01,
E	.5471E+00,	.5979E+00,	.6828E+00,	.1056E+01,	.1077E+01,
E	.5612E+00,	.5923E+00,	.6379E+00,	.8090E+00,	.8596E+00,
E	.5807E+00,	.6024E+00,	.6329E+00,	.7322E+00,	.7663E+00,

DISTRIBUTION RECORD FOR TREE-1081

Internal Distribution

- 1 - Chicago Patent Group - ERDA
9800 South Cass Avenue
Argonne, Illinois 60439
- 3 - A.T. Morphew
Classification and Technical Information Officer
Idaho Operations Office - ERDA
Idaho Falls, ID 83401
- 1 - H.P. Pearson, Supervisor
Technical Information
- 10 - INEL Technical Library
- 20 - Author

External Distribution

- 581 - UC-66d-GE - - Utilization Technology, TID-4500, R65

Total Copies Printed-616